**泰 山 学 院**

**本科毕业论文（设计）**

基于Spring Boot框架的文档搜索引擎

所 在 学 院 信息科学技术学院

专 业 名 称 软件工程

申请学士学位所属学科 工学

年 级 2020级

学生姓名、学号 莫玉珍 2020075170

指导教师姓名、职 称 刘婧 讲师

完 成 日 期 2024年5月15日

摘 要

本文阐述了一个基于Spring Boot框架和Elasticsearch技术的文档搜索引擎系统。该系统侧重于群组共享文件和全文文档搜索功能，旨在为企业和团队提供一个高效便捷的文档管理平台，解决文件过多、过于混乱、无法通过内容查询及时找到的问题。通过Spring Boot框架，系统实现了稳定的后端支持和出色的可扩展性。结合Elasticsearch强大的全文搜索能力，系统可以快速准确地检索大量文档，支持复杂的搜索条件，大大提高了搜索效率[1]。同时，系统还支持灵活的群组管理，实现团队成员之间的文件共享和协作，有效促进团队之间的沟通与协作。经过测试，系统功能丰富，易于扩展和维护，为企业和高校的日常工作提供了优秀的文档管理。

**关键词：**Spring Boot；ElasticSearch；文件在线预览；全文搜索引擎；

ABSTRACT

This article describes a document search engine system based on the Spring Boot framework and Elasticsearch technology. The system focuses on group-shared files and full-text document search functions, aiming to provide an efficient and convenient document management platform for enterprises and teams, solving the problems of excessive files, confusion, and inability to find content through content queries in a timely manner. Through the Spring Boot framework, the system achieves stable backend support and excellent scalability. Combined with Elasticsearch's powerful full-text search capabilities, the system can quickly and accurately retrieve a large number of documents, support complex search conditions, and greatly improve search efficiency. At the same time, the system also supports flexible group management, enabling file sharing and collaboration among team members, effectively promoting communication and collaboration between teams. After testing, the system has rich features, is easy to expand and maintain, and provides excellent document management for daily work in enterprises and universities.

**Keywords:** Spring Boot; ElasticSearch; Preview the file online; Full-text search engine;

目录

[1 引言 4](#_Toc166963055)

[1.1 研究背景 4](#_Toc166963056)

[1.2 国内外研究现状 4](#_Toc166963057)

[1.2.1 传统搜索引擎调研 4](#_Toc166963058)

[1.2.2 文档搜索产品调研 5](#_Toc166963059)

[1.3 研究内容和相关技术 6](#_Toc166963060)

[1.3.1 研究内容 6](#_Toc166963061)

[1.3.2 相关技术 6](#_Toc166963062)

[2 系统分析 7](#_Toc166963063)

[2.1 需求分析 7](#_Toc166963064)

[2.2 可行性分析 9](#_Toc166963065)

[3 系统设计 10](#_Toc166963066)

[3.1 系统架构设计 10](#_Toc166963067)

[3.2 系功能划分设计 11](#_Toc166963068)

[3.3 详细功能设计 12](#_Toc166963069)

[3.3.1 文档管理 4](#_Toc166963058)

[3.3.2 群组管理 5](#_Toc166963059)

[3.3.3 搜索模块 5](#_Toc166963059)

[3.4 数据存储设计 14](#_Toc166963070)

[3.4.1 数据库表设计 4](#_Toc166963058)

[3.4.2 ElasticSearch数据库设计 5](#_Toc166963059)

[4 系统实现 14](#_Toc166963071)

[4.1 开发环境搭建 15](#_Toc166963072)

[4.2 用户模块 15](#_Toc166963074)

[4.3 群组管理 15](#_Toc166963075)

[4.4 文件管理 15](#_Toc166963076)

[4.5 查询模块 15](#_Toc166963077)

[5 系统测试 15](#_Toc166963078)

[5.1 系统功能测试 15](#_Toc166963080)

[5.1.1 用户管理实现 4](#_Toc166963058)

[5.1.2 群组管理实现 5](#_Toc166963059)

[5.1.3 文件管理实现 5](#_Toc166963059)

[5.1.4 搜索模块实现 5](#_Toc166963059)

[5.2 测试总结 15](#_Toc166963081)

[6 结语 15](#_Toc166963082)

[参考文献 16](#_Toc166963083)

[致谢 18](#_Toc166963084)

1 引言

## 1.1 研究背景

随着信息技术的飞速发展，数据量的激增和信息的多样化使得文档管理和搜索成为各企业机构和高校工作中不可或缺的一部分。在日常运营和学术研究中，这些机构会产生大量的异构文件，包括报告、合同、研究论文、教案等，这些文档不仅是知识资产的重要组成部分，也是支撑业务流程和学术活动的基础。然而大量文档的存放使用传统的管理方式往往很难在需要查询文件时能够及时将文件筛选出来，根据实际调研和网络搜索总结出了如今的几个主要问题：

1）大多数文档都是以单独的形式存放，缺乏统一的管理和索引机制，这导致信息检索和查找变得困难且效率低下。当用户需要查找特定信息时，往往需要花费大量时间浏览和筛选文档，甚至可能错过重要的信息；

2）随着文档数量的不断增加，文档的管理和维护也变得越来越复杂。如何确保文档的完整性、安全性和可访问性，成为企业和高校必须面对的问题。传统的文档管理方式往往难以满足这些需求，导致信息资源的浪费和损失。

针对这些问题，设计一个高效、智能的文档搜索引擎显得尤为重要。文档搜索引擎可以通过对文档进行全文检索，快速定位到用户所需的信息，提高信息检索的效率和准确性[2]。同时，文档搜索引擎还可以对文档进行群组分类、分类查询等操作，进一步提升用户体验和文档管理的智能化水平。

## 1.2 国内外研究现状

## 1.2.1 传统搜索引擎调研

传统的搜索引擎产品主要分为以下几类：1）通用搜索引擎：国外的代表是谷歌，国内的代表是百度。这些搜索引擎获得的信息非常复杂。它们的搜索数据库依赖于互联网上所有网站的全文信息，更适合没有明确搜索意图的搜索者[3]；２）元搜索引擎:代表产品为360。这种搜索引擎本身不建立搜索数据库，而是将用户在本平台上发起的搜索请求转发给其他几个搜索引擎进行搜索，综合搜索结果返回给用户[4]。这类产品适合收集大量信息的搜索者;；３）垂直搜索引擎:这类产品专注于特定领域的搜索和细化[5]，也是近年来针对用户需求的新产品，比如携程专注于旅游机票和酒店的搜索，美团专注于美食和生活娱乐的搜索。此类搜索产品可以在特定领域为用户提供更专业、更全面的搜索结果和更好的搜索体验[6]，适合有特定搜索目标的搜索者;4)目录搜索引擎:代表Yahoo，这类产品以类似目录的方式对网站链接进行分类，不依赖关键词进行搜索，用户通过分类层进入目标网站。在信息迅速膨胀的现代，这种低效的检索方式逐渐被淘汰[7]。

传统搜索引擎在信息检索方面的便捷性是不言而喻的，但在企业和高校的信息检索中仍然存在局限性，因此全文文档搜索产品应运而生。

## 1.2.2 文档搜索产品调研

与其他信息不同的是，文件信息一般简洁、清晰、有插图，是传递信息的重要媒介。由于其各种优点，文档的数量也在不断增加，为了满足用户检索文档的需求，市场上出现了一系列相关的产品。目前与文档搜索相关的产品可分为以下几类:

１）桌面文档搜索系统：大多数桌面文档搜索系统都很全面，支持全文搜索，有些产品可以从文档中提取文本内容。国外Everything桌面搜索软件是比较有代表性的产品，它可以快速搜索文件名，找到文件所在的文件夹。但是它只能搜索文件名[8];Text Seek是国内具有代表性的桌面搜索软件，支持全文搜索，检索精度高，但检索速度相对较慢。但是桌面搜索系统产品的定位导致文档只能在本地进行搜索和查看，只适合个人用户，不适合需要随时随地查看和搜索文档的企业。

2)企业搜索引擎:目前已经出现了针对企业的搜索系统，如国内的智能搜索系统，这类搜索引擎侧重于搜索企业内各个方面的信息，包括网页、数据、文档等。虽然信息源很全面，但缺乏对文档等最重要、最常用的数据信息的功能优化[9]。

3)文档管理系统:这类产品虽然也提供文档内容的搜索功能，但更多侧重于文档的分类管理和存储，只提供简单的文件搜索功能，或者不提供全文搜索，或者不提供高级搜索，不能满足用户的搜索需求[10]。

通过国内外研究调研，结合用户需求，本文研究与实现一种基于Elasticsearch 的文档检索系统，设计并实现文档管理和文档搜索两大部分的功能，并引入自动KKFileView文档在线预览增强搜索并丰富搜索结果的展示。

## 1.3 研究内容和相关技术

1.3.1 研究内容

本研究旨在设计并实现一个基于ElasticSearch和Spring Boot框架的文档搜索引擎系统，以解决企业和高校在文档管理和搜索方面面临的主要问题。具体研究内容如下：

系统需求分析：深入调研和分析企业和高校在文档管理和搜索方面的实际需求，明确系统的功能需求和性能要求。

系统架构设计：利用Spring Boot框架，设计并实现一个稳定、可靠、可扩展的文档搜索引擎系统。系统架构将包括前端用户界面、后端服务层、数据访问层等多个部分，以确保系统的完整性和可维护性。

文档管理功能实现：开发文档管理模块，实现文档的上传、存储功能。通过构建合理的文档存储结构，确保文档的完整性、安全性和可访问性。

文档搜索功能实现：利用ElasticSearch技术，构建文档的索引库，并实现高效的全文检索和高级搜索功能。通过优化搜索算法和索引策略，提高检索的效率和准确性[11]。

自动文档预览功能实现：引入自动KKFileView文档在线预览技术，为用户提供文档内容的快速预览功能。通过在线预览，用户无需下载文档即可快速了解文档内容，提升用户体验。

1.3.2 相关技术

Spring Boot框架：作为系统的基础框架，Spring Boot提供了快速搭建企业级应用的能力。它简化了Spring应用的初始搭建和开发过程，通过约定优于配置的方式，使开发者能够更专注于业务逻辑的实现[12]。

ElasticSearch：ElasticSearch是一个基于Lucene的开源搜索引擎，它提供了一个分布式、多租户能力的全文搜索引擎[13]。通过构建文档的索引库，ElasticSearch可以实现高效的全文检索和高级搜索功能[14]。在本研究中，ElasticSearch将作为文档搜索引擎的核心技术，用于实现文档的快速检索。

数据库技术：根据需要，本研究将使用适当的数据库技术MySQL来存储和管理系统数据，确保数据的安全性和可靠性。

前端开发技术：采用现代的前端开发技术HTML5、CSS3、JavaScript、Bootstar等构建用户友好的前端界面，提供良好的用户体验。

2 系统分析

## 2.1 需求分析

本文档搜索引擎的目标是为用户提供一个功能丰富的文档管理和搜索平台，旨在满足企业和高校对文档管理的需求。系统将实现用户和管理员的登录注册功能，为用户提供便捷的身份认证和个性化设置。如图2-1所示，用户可以通过系统进行文档上传和搜索操作，以便快速查找和共享所需文档。此外，系统还支持群组的创建和管理，用户可以根据团队或项目需求创建不同的工作群组，并在群组内进行文档共享和讨论。

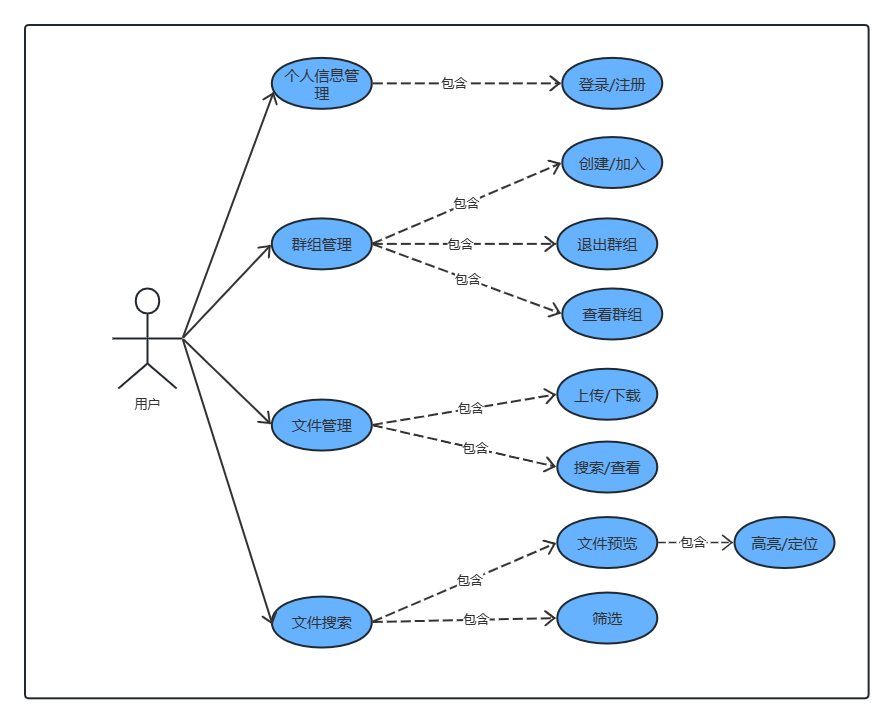


图 2‑1 用户用例图

为了确保系统的日常工作能够得到保证，增加了管理员角色来维护系统，如图2-2所示，管理员可以对所有用户，群组和文件有删除和添加的权限，通过发布文件来显示系统的公告。

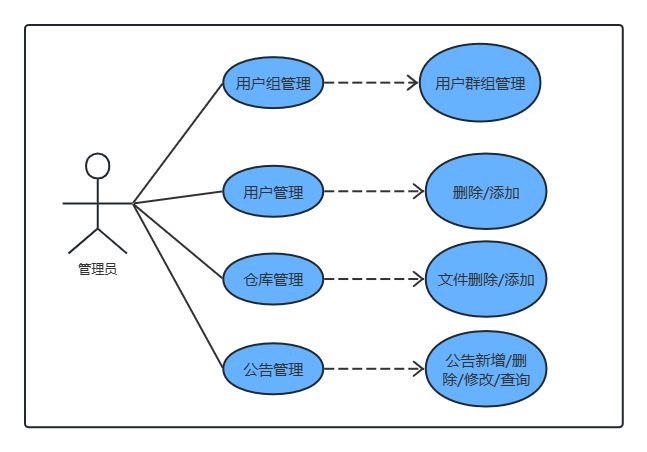


图 2-2 管理员用例图

通过提供这些核心功能，系统旨在帮助企业或团队更高效地管理和查找文档，提升工作效率和协作效果。

## 2.2 可行性分析

在技术方面，选择基于 Spring Boot 框架进行开发，该框架具有成熟的技术生态系统和丰富的社区支持，能够快速搭建稳定可靠的应用。同时利用现有的开源组件或工具，如 Elasticsearch 用于文档索引与搜索[15]，Spring Security 用于身份认证与权限控制，大大降低了开发成本和风险。另外，还将采用前后端分离的架构，利用现代化的前端框架Bootstar开发用户界面，提升用户体验。

在经济方面，采用成熟的开源技术和组件可以降低系统的开发和维护成本。通过灵活的部署方案，如使用云服务提供商提供的弹性计算和存储服务，可以根据实际需求灵活调整系统的规模，降低运维成本并提高系统的稳定性和可靠性。

在实现方面，前端将设计简洁直观的用户界面，使用户能够快速上手并熟练使用系统的各项功能。同时提供详细的用户手册和技术文档，帮助用户解决常见问题和疑问。此外，建立完善的技术支持和售后服务体系，及时响应用户反馈并提供技术支持，确保系统的稳定运行和用户满意度。

综上所述，从技术、经济和实现等方面分析，文档搜索引擎系统具有较高的可行性和可实施性，能够有效地满足企业或团队对文档管理和搜索的需求，提高工作效率和协作效果。

3 系统设计

## 3.1 系统架构设计

基于上述需求分析，本文旨在构建一个功能完善、易于使用、安全可靠的文档管理和搜索平台。系统将采用模块化设计，确保各个功能模块之间的松耦合和高内聚，以便后续的维护和扩展。同时，遵循用户友好的设计原则，提供直观易用的用户界面和丰富的功能选项，满足企业和高校对文档管理的多样化需求。

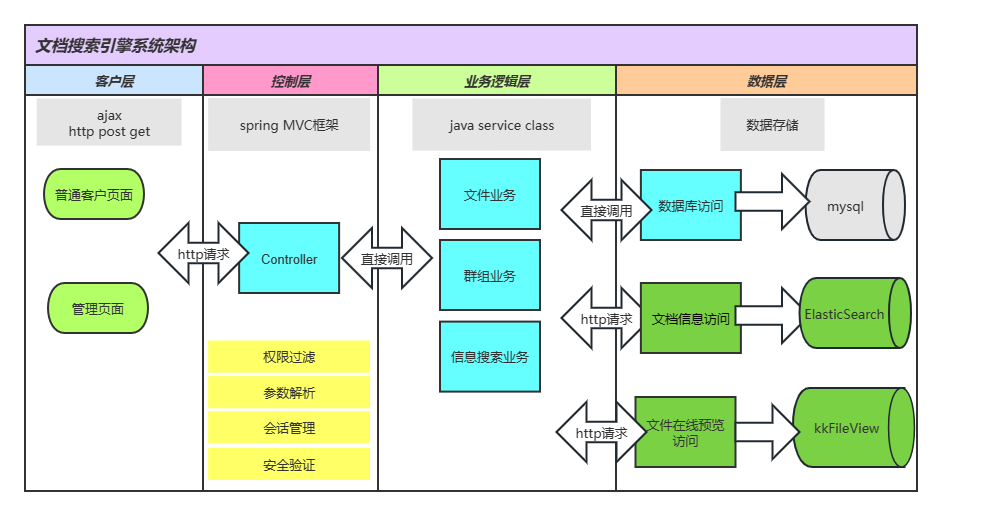


图 3-1 系统架构图

如图3-1，系统采用前后端分离的架构，前端主要负责用户界面的展示和交互逻辑的处理，后端则负责业务逻辑的实现和数据存储的管理。前后端之间通过API接口进行通信，确保数据传输的安全性和稳定性。

## 3.2 系统功能划分设计

针对用户需求，文档搜索系统设计了权限模块、管理模块和搜索模块三大部分，如图3-2展示了各类模块的主要功能。

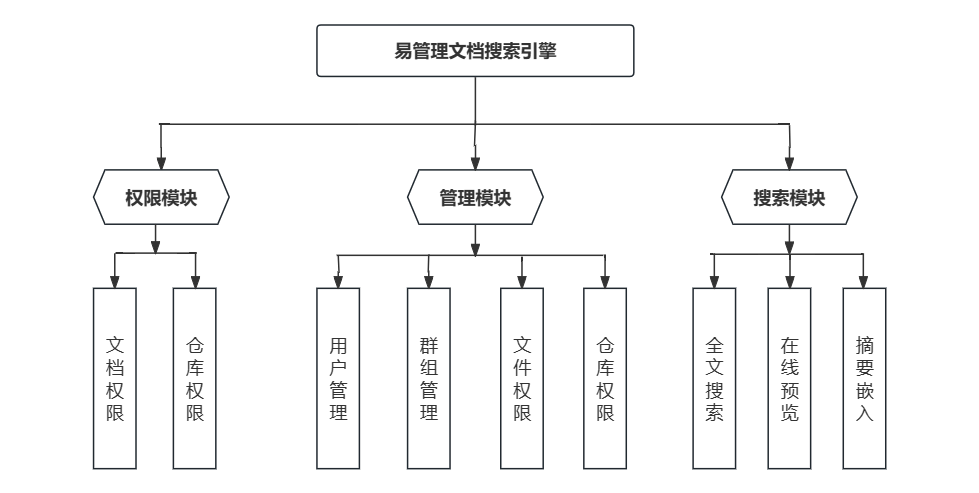


图 3-2 系统功能结构图

权限模块负责管理和控制用户对系统资源的访问权限，以确保系统的安全性和数据保护。权限模块包括以下子模块：

1）文档权限负责设定和管理用户对文档的访问权限，包括读、写、修改等操作。通过细致的权限设置，可以防止未经授权的用户访问敏感信息。文件仓库权限管理用户对文档仓库的访问权限。仓库是文档的存储单位，通过对仓库的权限控制，可以限制用户对特定文档集合的访问；

2）管理用户账号的创建、修改、删除等操作，以及用户的基本信息，如用户名、密码、邮箱等。同时，用户管理还负责用户角色的分配，以便在权限设置时能够方便地引用角色；

3）群组管理：将用户分组，并为每个组分配特定的权限。群组管理可以简化权限设置过程，提高管理效率。

管理模块负责系统的日常运营和维护，包括文件管理和系统监控等任务。管理模块包括以下子模块：

1）文件管理负责文档的上传、下载、删除等操作的同时还可以对文档进行分类、打标签等操作，以便于文档的检索和管理；

2）监控系统的运行状态，包括服务器负载、网络带宽、磁盘空间等关键指标。当系统出现异常时，能够及时发出警报并采取相应的措施。

搜索模块是文档搜索系统的核心功能之一，它允许用户根据关键词或条件查询系统中的文档。搜索模块包括以下功能：

1）全文搜索：对文档中的全部文本内容进行搜索，确保用户可以快速地找到相关信息；

2）在线预览：提供文档的在线预览功能，使用户可以在不下载文档的情况下浏览文档内容；

3）摘要嵌入：对于长文档，可以提供摘要功能，帮助用户快速了解文档的主要内容。同时，摘要还可以嵌入到搜索结果中，提高搜索结果的可读性。

## 3.3 详细功能设计

3.3.1 文件管理

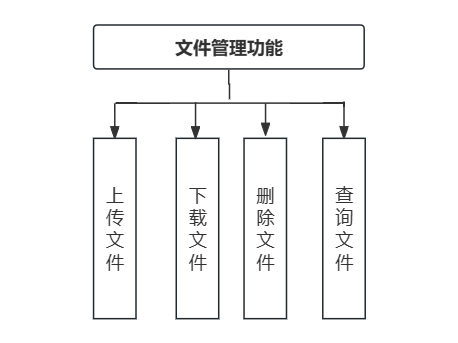


图 3-3 文件管理结构图

用户可以通过上传文件功能将需要管理的文件添加到系统中，随后用户可以通过查询文件内容信息后筛选出有需求的文件从而在系统中获取所需文件并下载。若用户对系统中的文件不再需要时，用户可以通过删除文件功能将其从系统中移除。为了用户能够更加确定查询的结果是否是自己需要的文件，在查询文件结果展示中，系统为用户提供了一个在线预览文件的接口，方便用户在不需要下载文件的情况下也能查看文件内容。详细功能如图3-3。

3.3.2 群组管理

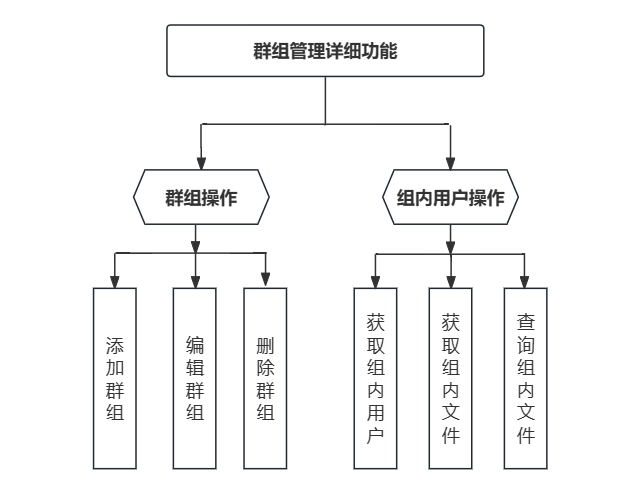


图3-4 群组管理结构图

在群组操作模块中包含了群组管理的基础操作，用户可以直接进行创建，添加，删除群组的操作等最基本的任务，同时本系统还支持群组内部用户的管理，作为群组的普通用户，可以获取到组内成员的列表信息，随后可以进一步执行与组内用户相关的操作，如获取组内文件或查询组内文件。而群组创建用户可以对群组成员的增加和删除进行操作。详细功能如图3-4

3.3.3 搜索模块

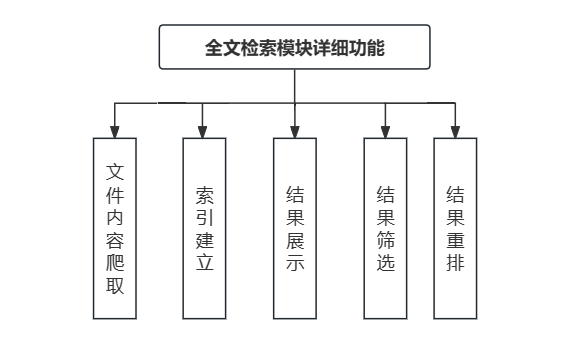


图 3-5 文件管理结构图

为了实现全文文档搜索功能，本系统使用了ElasticSearch来存储文件的内容同时对文件的内容进行自动检索和索引，同时还使用了MySQL存储文档的主要信息。在显示用户进行文件的搜索页面时，还对搜索到的文件信息组进行筛选，根据ElasticSearch结果匹配度从高到低的顺序排序并返回到前端页面，最后使用kkFileView完成对文档的在线预览。详细功能如图3-5。

3.4 数据存储设计

3.4.1 数据库表设计

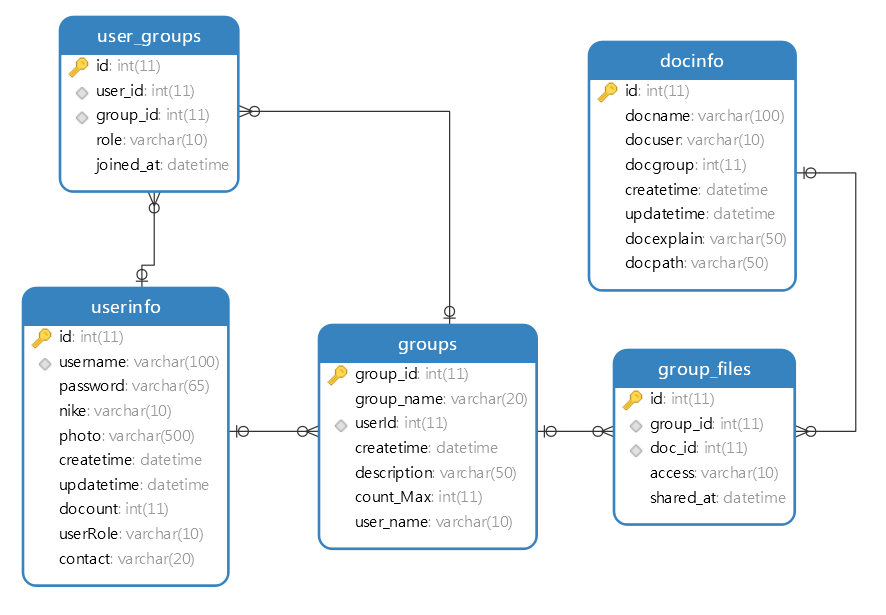
文档知识搜索引擎的ER图如图3-6所示，包括用户（userinfo），文件（docinfo）、群组文件（groups\_file）、群组（groups）、用户组成员（user\_groups）共5张表。

图 3-6 数据库ER图

在这些数据表中，用户表和群组表相关联，一个用户可以有多个群组，一个群组可以有多个用户；用户和文件相关联，一个用户上传多个文件，而一个文件对应一个作者，只能由该作者删除；文件和群组相关联，文件可以被放置在多个群组中，而一个群组也可以存在多个不同作者，已经同一个作者的多个文件。

3.4.1 ElasticSearch数据库设计

系统会解析用户上传的文档，取到文档的文件信息和内容信息，并建立索引，而非关系型数据存储ElasticSearch是通过Json格式的Mapping映射来定义索引文档的字段信息和类型等。表3-1解释了本文中使用ElasticSearch建立Mapping中主要字段的相关信息。

表 3-1



4 系统实现

## 4.1 开发环境搭建

系统整体采用B/S架构，分为前端界面和后端服务设计。系统前端采用Bootstrap框架，后端采用Java语言下的SpringBoot框架。系统整体采用了分层设计，Controller层用于接收前端的请求并进行初步处理，并将其转发到Service层用于处理具体的业务逻辑；如若涉及到数据库相关的部分，则转发到Entity层与数据库进行交互；Service层的处理结果会返回Controller层，并使用定义好的统一返回类型AjaxResult进行数据封装，提交到前端的View层。在长文本摘要实现方面，文本摘要部分通过Http请求与系统其他部分进行交互。因此根据以上要求，在完成本文系统要进行以下开发环境的搭建：

1. 后端代码编写使用IDEA2021.3.2 版本的编译器，本文系统使用的是SpringBoot2.7版本，因此Java语言选择对应的1.8版本进行对后端逻辑的实现，同时使用Mybatis对MySQL的数据进行操作；
2. 要完成对文件的在线预览，本文采用了kkFileView开源项目来完成。因此要在社区上拿到该项目的源码并在编译器上运行成功且配置好与本系统进行的接口；
3. 对文件进行分析与索引都交给ElasticSearch分布式数据库完成，因此要将ElasticSearch下载到本地并配置本地端口，由于ElasticSearch内置的分词器不能满足我们文件内容的检索要求，要将IK分词器嵌入ElasticSearch中；
4. 前端使用了VSCode编译器可以便捷我们进行前端页面的编写，同时使用Bootstrap框架使前端页面变得更加简洁美观。

## 4.2 用户模块实现

用户模块主要使用user相关的接口进行交互，具体接口见下表4-1

表 4-1

如图4-1所示，系统会检测当前进入网站的用户，判读是否已经登录，若为游客身份，进行某些如文件上传操作时会自动进入到注册页面，若是已经注册的用户可以跳转到登录页面，用户使用账号和密码进行登陆，前端获取到数据传到后端与数据库进行比较，若是数据库信息和传来的数据相同则登录成功，密码错误则返回登录页面。

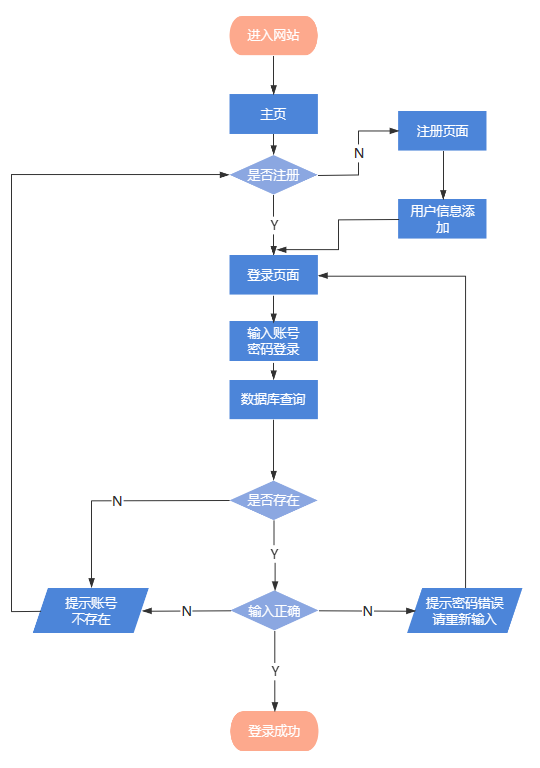


图 4-1

## 4.3 群组管理实现

群组管理模块主要使用group相关的接口进行交互，具体接口见下表4-2。

表格 4-2



如图4-2所示，用户会在列表页面上看到全部已经公开的群组，同时也可以进行创建群组的操作，在群组列表页面系统会自动判断是否已经加入群组页面，若干用户已经加入则不显示加入群组按钮；选择创建群组会进入创建群组信息页面，信息上传后会在后端进行与已有的群组进行比较，如果群组已经创建则返回到群组列表页面，可以再次选择加入群组，如果群组未创建则在数据库信息中添加刚创建的群组。

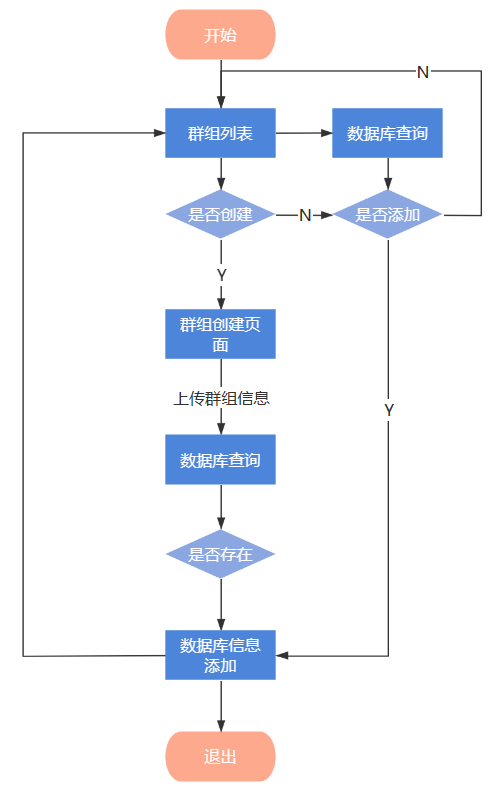


图 4-2 群组创建流程图

## 4.4 文件管理实现

文件管理模块主要使用file相关的接口进行交互，具体接口见下表4-3。

表 4-3



如图4-3所示，用户上传群组时会显示文件信息编辑页，同时系统会获取到当前用户session返回后端，后端根据传来的信息将文件进行分类与权限管理，看当前文件是否已经上传过，同时会判断文件大小是否能够存储进当前数据库中，当条件都满足时显示文件上传成功并且更新数据库信息。

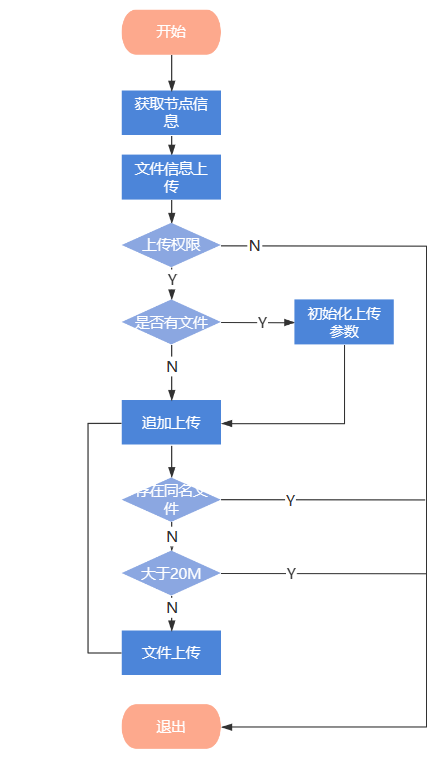


图 4-3 文件上传流程图

## 4.5 查询模块实现

如图4-4所示，在对文档建立完倒排索引后，当用户输入查询语句时，后端会对其进行分词处理，并将其转换成对应ElasticSearch的结构化查询语句。在查询时还会通关键词过滤掉当前登录用户无权限的文档，只在其有权限的文档内进行匹配查询，最后对查询结果进行处理与展示。

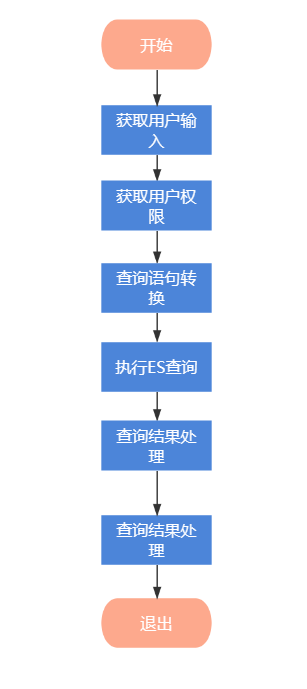


图 4-4 文件信息搜索流程图

5 系统测试

## 5.1 系统功能测试

5.1.1 用户模块测试

测试目的：验证用户是否能够在主页进行登陆和注册页面的跳转，能否通过注册和登录进入系统并进行相关身份验证操作。

测试方法：在游客状态进入平台主页页面，查看主页顶部导航栏状态是否存在注册按钮，然后点击需要用户身份的操作，判断是否能够进行，若是游客状态下顶部导航显示注册按钮以及点击文件上传时弹出需要登录的提示框。

预期结果：游客状态下显示注册按钮不能进行文件上传操作，点击注册按钮会跳转到注册页面，注册完信息后进入登录页面验证信息无误再次返回主页，此时无注册按钮，并且能够进行文件上传的身份验证的操作。

测试结果：符合预期结果，部分结果如图5-1，5-2所示。

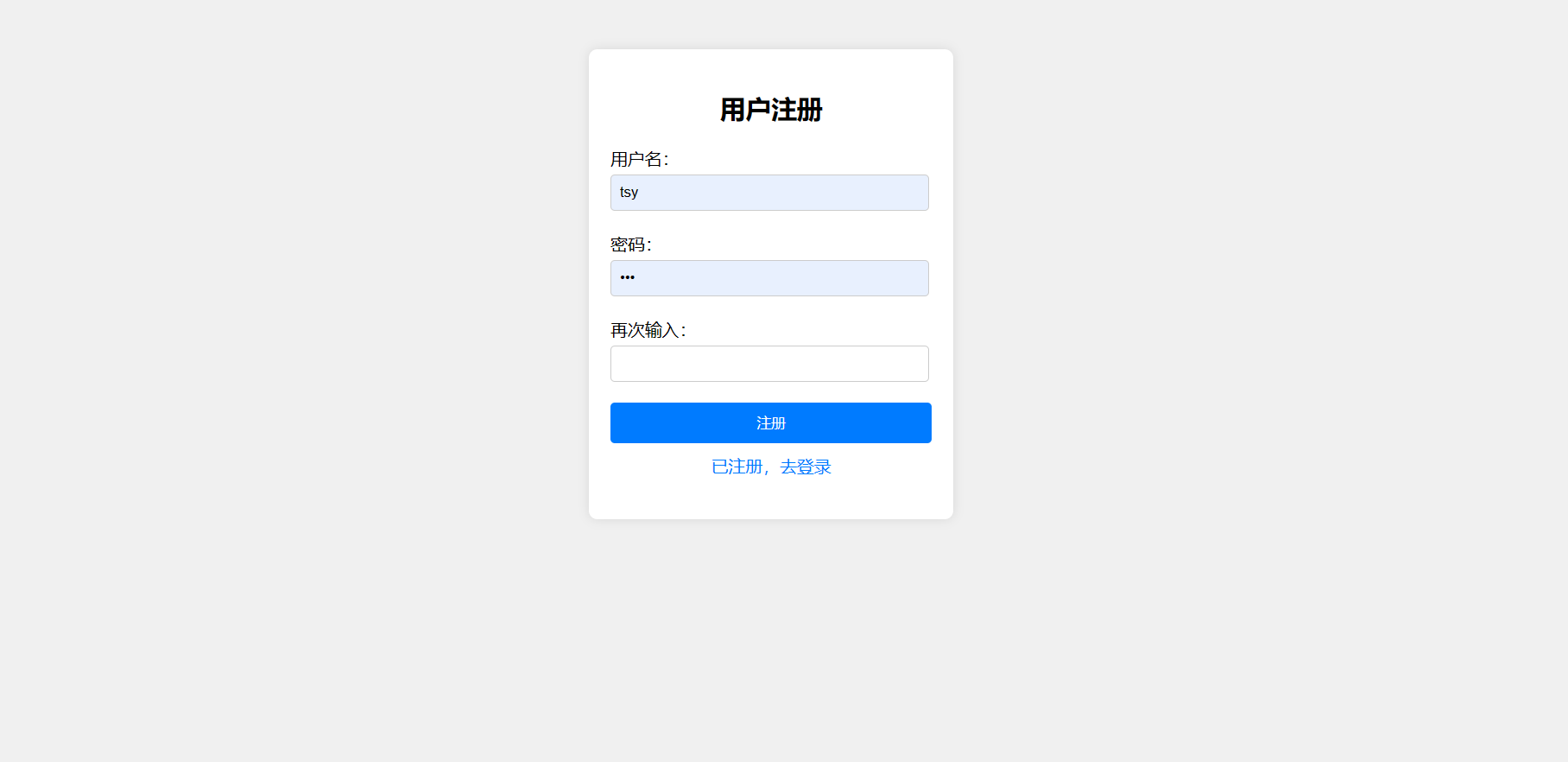


图 5-1 用户注册界面



图 5-2 登录界面

5.1.2 群组管理测试

测试目的：验证用户能否进行添加，创建，删除群组，已经在创建好的群组下删除群组成员，并且能够成功显示群组成员信息和文件信息。

测试方法：用户A点击创建群组功能，创建两个群组后切换用户B，添加其中一个群组，刷新页面查看已添加的群组，查看成员列表，之后再进行创建群组的操作，创建群昵称与加入的群组相同，判断能否创建，完成后切换回创建群组的用户A，查看群组是否有成员加入，删除没有用户加入的群组。

预期结果：用户B不能再次创建用户A已经创建好的群组，会显示群组已经创建，用户A会看到用户B已经加入其中一个群组，并且能够对该用户进行删除退出群组功能，点击删除没有用户加入的群组后会显示删除群组成功。

测试结果：符合预期结果，部分结果如图5-3所示。

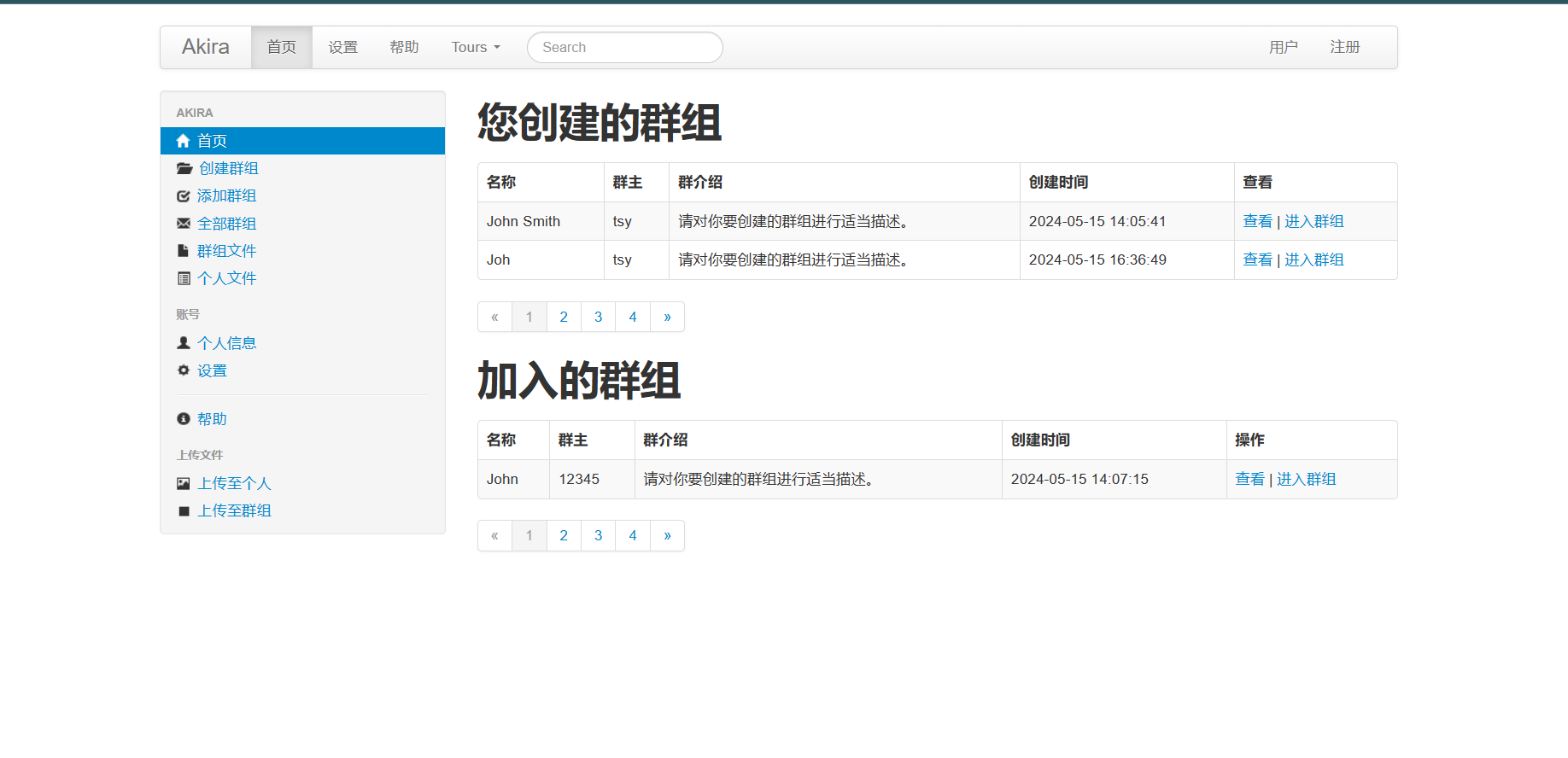


图 5-3 群组列表界面



图 5-4 群组成员列表界面

5.1.3 文件管理测试

测试目的：验证用户能否成功上传文件至群组和个人文件，并且能够对已上传的文件进行删除和预览。

测试方法：用户上传文件至个人，再次上传文件至群组中，在个人文件列表中查看是否存在刚刚上传的文件，在群组文件列表中查看是否存在上传的文件。

预期结果：用户上传完两个分别不同类别的文件后，群组内所有用户都能查看到上传的文件，用户能在个人文件列表中看到刚上传的文件，MySQL以及ElasticSearch数据库都能查询到上传的文件信息。

测试结果：符合预期结果，部分结果如图5-5所示。

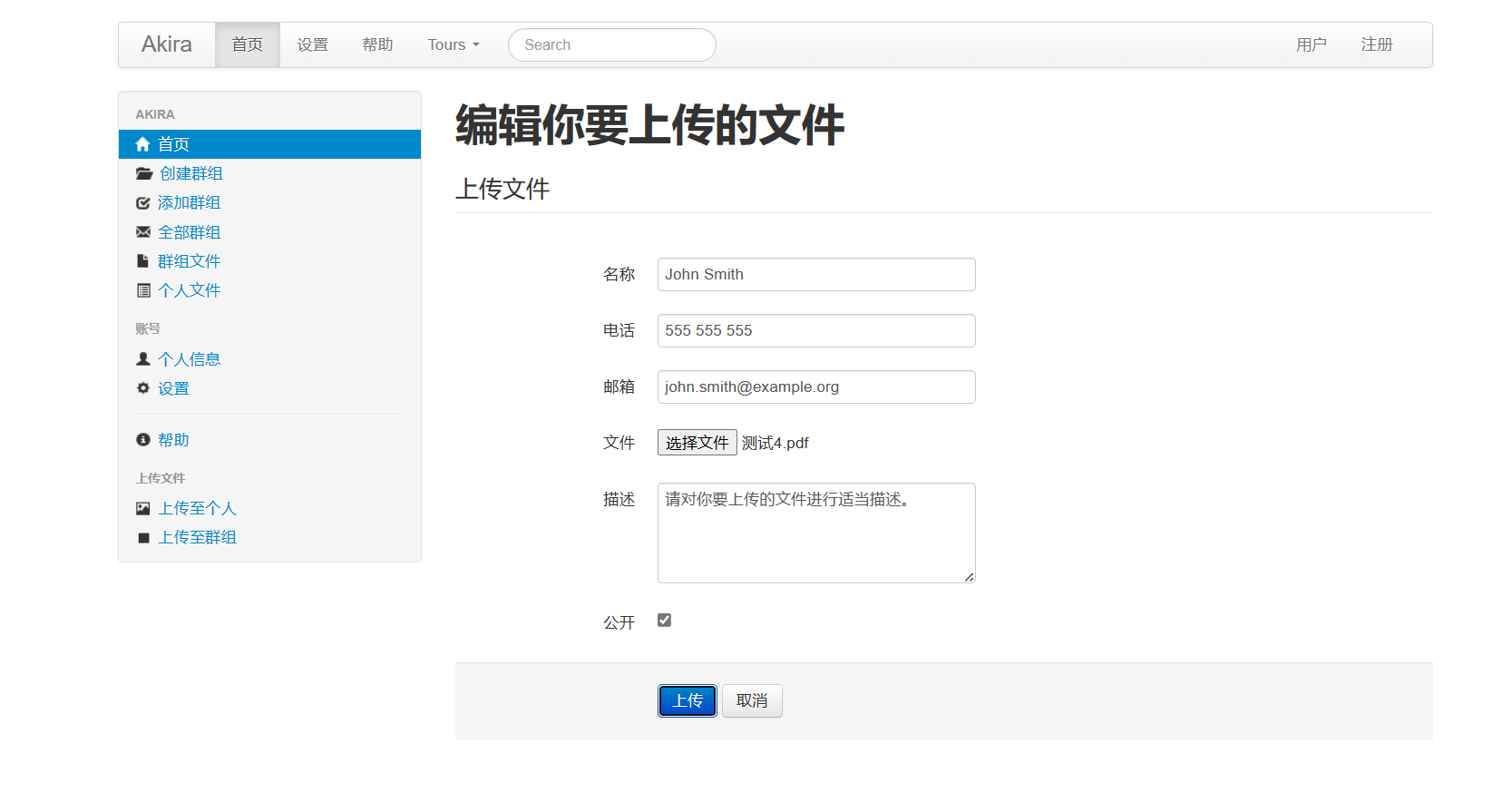


图 5-6 文件上传界面

5.1.4 搜索模块测试

测试目的：验证用户能否在群组以及所有公开文件中搜索到与关键词匹配的文件，同事能看到关键词高亮信息。

测试方法：用户在主页的搜索所有公开文件中进行搜索，之后在群组界面中进行搜索。

预期结果：在主页搜索后能够看到所有跟关键词有关的公开文件，同时还能看到文件上传时间，文件作者，在群组搜索后能够看到群组成员发布的有关关键词的所有文档，两个方式搜索的文档展示效果内容都是以关键词所在的句子部分展示，同时关键词会被高亮显示在句子当中。

测试结果：符合预期结果，部分结果如图5-7,5-8所示。



图 5-7 搜索展示界面



图 5-8 文件预览界面

5.2测试结果

经过测试，可以看出该文档搜索引擎在前后端的数据传输通畅，后端逻辑清晰不苟，前端能够很好展示了页面效果，很好的展示了文件信息和内容。测试结果表明，本次测试的全文文档搜索引擎在基本功能方面表现正常，符合预期要求。

6 结语

本文对现有文档搜索市场和相关产品进行调研，设计并实现了符合企业和高校需求的文档搜索系统，并对用户发布的文件进行了群组分类处理，完成了文档搜索结果高亮展示和文档在线预览。在研究与实现过程中也注意到以下有待改进的工作：

1. Elasticsearch 可能会消耗大量的内存和磁盘空间，特别是当索引数据量很大时。如果不加以控制，大量的索引和查询可能会消耗过多的系统资源，为此本系统应该对 Elasticsearch 进行适当的配置和优化，以提高性能和资源利用率。
2. Elasticsearch 是一个最终一致性系统，这可能会导致在并发写操作时出现数据不一致的问题。如果没有实施适当的数据校验和备份策略，可能会面临数据丢失或损坏的风险。本系统应该设计具有高伸缩性和可用性的系统架构，以确保系统在高并发或大数据量下的稳定性。

参考文献

1. 郭向民,袁许龙,朱洛凌.基于Scrapy和Elasticsesarch的网站敏感词检测系统[J].网络空间安全,2024,15(01):70-75.
2. 董元和,贾炎,朱勇,等.基于ElasticSearch分布式搜索引擎的信息检索方法研究[J].湖北师范大学学报(自然科学版),2023,43(04):56-61.
3. 司纯.基于深度学习的企业文档知识搜索系统的研究与实现[D].北京邮电大学,2023.DOI:10.26969/d.cnki.gbydu.2023.001365.
4. 朱艳芳,赵蕾,王琪.针对Lucene搜索引擎的算法研究[J].软件,2023,44(05):161-164.
5. 黄天伟.基于文本分析技术的内容农场过滤系统的设计与实现[D].电子科技大学,2023.DOI:10.27005/d.cnki.gdzku.2023.001307.
6. 王贺.法律搜索引擎中倒排索引构建技术的研究及应用[D].辽宁科技大学,2023.DOI:10.26923/d.cnki.gasgc.2023.000246.
7. hoshal A .Aerospike adds connector for Elasticsearch to run full-text queries[J].InfoWorld.com,2023,
8. aryam S ,Abolghasem S ,SooMi C , et al.AR Search Engine: Semantic Information Retrieval for Augmented Reality Domain[J].Sustainability,2022,14(23):15681-15681.
9. 杨晓波.基于Lucene的文档数据自动解编检索系统的设计与实现[C]//西安石油大学,陕西省石油学会,北京振威展览有限公司.2022国际石油石化技术会议暨新能源及节能技术国际会议论文集.大庆油田勘探开发研究院;,2022:9.DOI:10.26914/c.cnkihy.2022.053249.
10. 刘银.基于元搜索引擎的网络信息智能检索系统设计[J].长江信息通信,2022,35(06):156-158.
11. 朱艳.轻量级检索服务的词干提取与相关排序优化研究[D].桂林电子科技大学,2022.DOI:10.27049/d.cnki.ggldc.2022.000204.
12. Fábio M ,Matt P ,N R A T , et al.Search and sequence analysis tools services from EMBL-EBI in 2022.[J].Nucleic acids research,2022,50(W1):W276-W279.
13. 高剑.基于Lucene检索工具的科技查新检索方法设计[J].集成电路应用,2022,39(04):114-115.DOI:10.19339/j.issn.1674-2583.2022.04.044.
14. Daqing W .Digital Archive Management Based on Lucene Full-text Search Engine[J].Journal of Physics: Conference Series,2021,2074(1):
15. Wentao X ,Haoyu C ,Yidong H , et al.Full-text search engine with suffix index for massive heterogeneous data[J].Information Systems,2022,104

致谢

锲而不舍，金石可镂。在我写论文的这段时间里，我感到对学习的知识和能力的磨练。这促使我以最快的速度投入到撰写中去，通过这次，我学到了很多知识，能够灵活掌握课本所学的理论，应用到开发中，我的各方面水平都有了较大的提升。每一次的经历对我来说都是一种成长，每一次的成功都给我带来巨大的满足感。在这个过程中我积累的经验将对我以后的学习和生活产生很大的帮助。

谢谢我的论文指导老师刘婧老师。这篇论文是在老师的指导下完成的，感谢为我提供帮助的同学们。在各方的帮助下，我的毕业论文完成了。

最后，感谢学校所有的老师。在过去的四年里，你们无私的奉献精神和严谨的学术态度使我受益匪浅。最后，我感谢学校给了我宝贵的学习机会，让我开始新的生活，并向更好的未来迈进。