



中国地质大学（北京）

## 软件工程导论 B 报告

选    题：地球化学实验室数据管理平台

指导老师：管青

组    长：王俊博 1004211127

成    员：沈    蓉 1004211101

付凯威 1004211126

李炯炫 1304231102

邮    箱：junbowang@email.cugb.edu.cn

日    期：2024.04.23

# 目 录

## 目录

1. 选题.....	3
1.1 选题背景 .....	3
1.2 国内外发展及研究现状 .....	3
1.3 研究意义 .....	6
1.4 可行性分析 .....	7
2. 系统关键技术.....	8
2.1 前端 .....	8
2.2 后端 .....	8
2.3 数据库 .....	9
2.4 设计和管理工具 .....	9
3. 系统需求分析.....	10
3.1 需求定义 .....	10
3.2 系统流程分析 .....	12
3.3 应用环境 .....	14
4. 系统用例分析.....	15
4.1 系统用例图 .....	15
4.2 系统用例文档 .....	15
5. 数据库.....	21
5.1 数据库关系图 .....	21
5.2 数据库核心结构 ER 图 .....	21
5.3 数据库结构解释 .....	22
5.4 数据库字典 .....	22
6. 系统架构.....	23
6.1 架构详细 .....	23
6.2 系统类图 .....	24
6.3 系统详细设计 .....	25
7. 系统展示和说明.....	26
7.1 原则 .....	26
7.2 用户界面 .....	28
8. 白盒测试.....	35
8.1 注册功能测试 .....	35
8.1 登录功能测试 .....	37
9. 黑盒测试 .....	38
9.1 注册功能测试 .....	38
9.2 登录功能测试 .....	40
10. 压力测试 .....	41
10.1 目标和指标 .....	41
10.2 场景设计 .....	41
10.3. 数据准备 .....	41

## 1. 选题

### 1.1 选题背景

长期以来，地球化学实验室一直依赖于 Excel 等表格工具来处理实验数据。尽管这些工具在使用上相对简便，但随着科学研究的不断深入和数据规模的急剧增加，这种传统方式已经显露出一系列的不足。

近年来，随着数据驱动型地球科学问题的兴起，许多新工具也应运而生。其中，Python 框架 Geochemistry  $\pi$  提供了丰富而易于使用的接口，极大地拓展了那些编程基础相对薄弱的地球化学家们的研究范围。

因此，我们决定着手建立一套全新的地球化学实验室数据管理平台，旨在为实验室提供一种简便易行、易于部署、并支持数据驱动型研究的全面解决方案。通过这一平台，我们能够更为轻松地处理庞大的实验数据，提高数据的共享和可访问性，同时也为科学家们提供了更强大的工具，帮助他们深入开展数据驱动型研究，推动地球科学领域的创新。这一努力旨在提升实验室内部的协作效率，提高数据管理的整体水平，为地球化学研究创造更为便捷而创新性的研究环境。

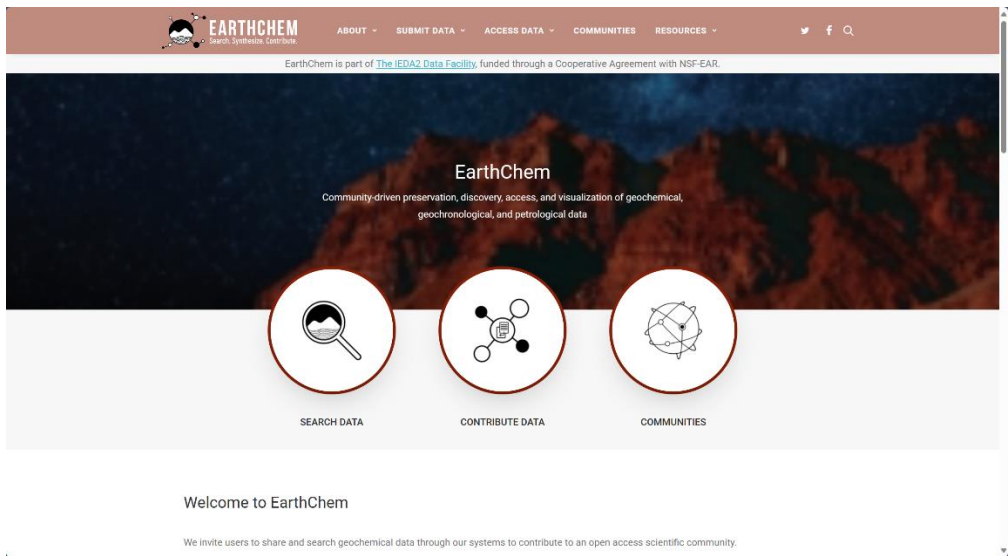
### 1.2 国内外发展及研究现状

#### ➤ EarthChem

EarthChem 数据库是一个全球性的地球化学资源，旨在整合和提供丰富的地球化学数据，以支持科学家、研究人员和学生在地球科学领域的研究和教育。该数据库收录了来自全球各地的地球样品数据和地球观测数据，涵盖了岩石、矿物、土壤、沉积物等各种地质样本的化学分析数据，以及地球表层和深部地球化学特征的观测数据。这些数据不仅包括各种元素和同位素的分析结果，还包括地球化学特征的描述、地球化学反应的数据等，为地球科学研究提供了宝贵的信息和参考资料。

EarthChem 致力于为用户提供便捷的数据访问和下载接口，使他们能够轻松获取所需的地球化学数据。科学家可以利用这些数据来探索地球的成因、地质过程、地球化学循环等重要问题，推动地球科学领域的进步和创新。同时，EarthChem 还为教育工作者提供了丰富的教学资源，可用于大学课程、研究生培训等教学活动，促进学生对地球科学的理解和认识。

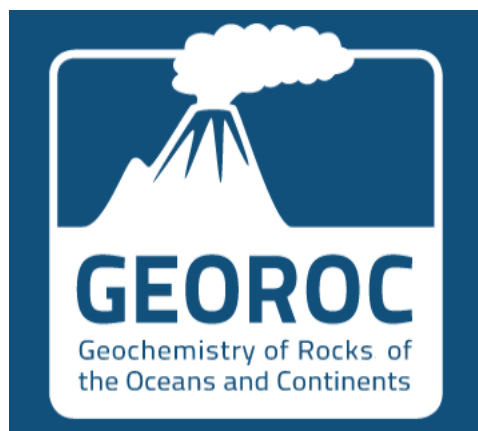
作为全球地球化学数据的综合性平台，EarthChem 的愿景是成为地球科学领域的重要支撑，推动地球化学研究和教育的发展，促进地球科学的进步和创新，从而更好地理解和保护我们的地球。



➤ **GeoRoc: 全球汇聚边界岩浆岩数据库**

GeoRoc 是一个全球性的地球科学数据库，旨在为科学家、研究人员和学生提供丰富的岩石地球化学数据和相关元数据。该数据库收集了来自各种地质环境的岩石样品的地球化学分析数据，包括火成岩、沉积岩、变质岩等不同类型的岩石。GeoRoc 的数据涵盖了各种元素的含量数据，以及岩石的地球化学特征、成因信息等。这些数据对于理解地球内部构造、地质过程以及岩石形成演化具有重要意义。数据库提供了丰富的元数据支持，包括岩石样品的地质背景、采样地点、采样日期等信息，帮助用户更好地理解数据的背景和可靠性。GeoRoc 的数据对科研人员、学生和公众开放获取，用户可以通过数据库平台查询、浏览和下载数据，支持科学研究和教育活动。通过提供

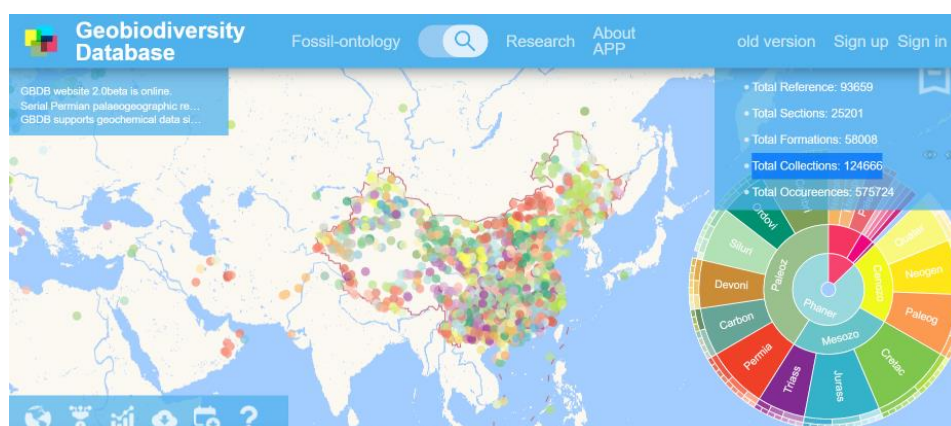
这些数据资源，GeoRoc 数据库为地球科学研究提供了重要的支持，促进了对地球内部结构、地质过程和地球化学演化的深入理解。



#### Current content of GEOROC:

- 21,370 papers
- 640,100 samples
- 2,402,280 analyses
- 35,116,090 single data values

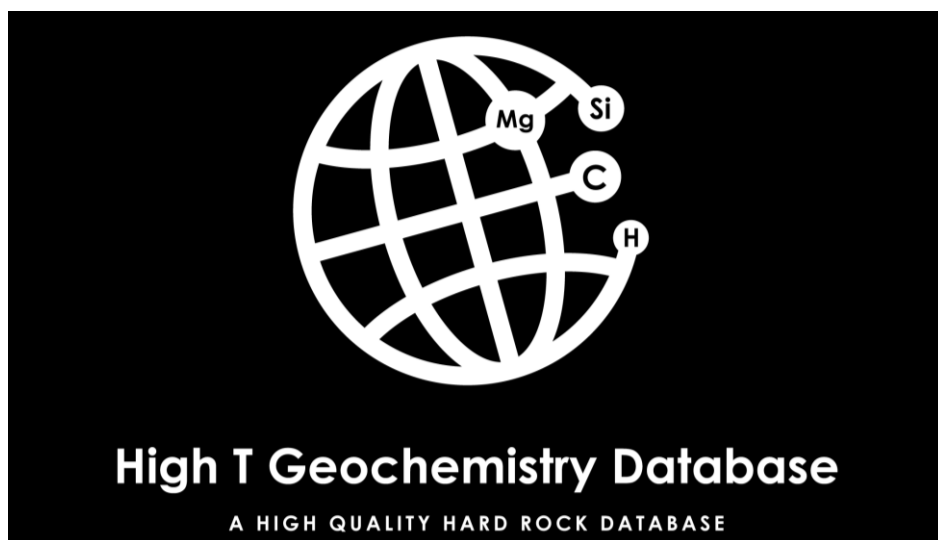
- **Geobiodiversity Database:** 南京大学古生物化石数据库  
由中国科学院南京地质古生物研究所于 2007 年成立的基于地层剖面的古生物学和地层学的综合数据库。



- **HTG Benchmark Database:** 深时数字地球（DDE）子项目  
由浙江大学开发的高温地球化学标准数据库提供了人工清洗后的高质量数据，可用于数据驱动型的地球科学问题研究。统计方法和机器学习算法在网络基础设施获取的数据上的快速发展，为揭示地球固体演化和化学提供了真正的机会。然而，目前的网络基础设施由原始的高温地球化学数据组成，其中存在缺失类别、相当比例的错误，包括年龄信息，以及可能与出版物不一致的化学组成。这些非预期的错误主要由以下原因引起：（1）手工数据输入错误；（2）缺乏发布高温地球化学数据的标准，以及用于将表格数据转换为可读文档的光学字符识别技术的限制。此外，岩石和矿物之间缺乏固有关系。在 20 位地球化学家的共同努力下，ZJU\_Earth\_Data team 构建了一个

高温地球化学的基准数据集。到目前为止，已经手动检查和根据 FAIR 原则（可查找、可访问、可互操作、可重用）校正了 20 万个岩石和矿物数据。此外，数据库提供了从网络基础设施下载的原始数据以及经过数据清理后的对应数据，可用于检查数据过滤算法的有效性。因此，这 20 万个样本包括：

（1）原始-清洁数据对；（2）矿物-矿物和矿物-岩石关系。网络门户提供了查询功能，可搜索特定的地球化学数据，以及匹配功能，可找到在相同形成条件下生成的岩石-矿物组合和矿物-矿物对。



### 1.3 研究意义

#### ➤ 数据整合与共享

当前地球化学研究领域积累了大量的样品分析数据，这些数据分散在各个实验室和科研项目中，缺乏有效的整合和共享机制。通过构建一个集中的地球化学数据管理平台，能够实现数据的标准化存储、统一管理和便捷访问，促进跨学科、跨地域的数据共享，极大提升数据利用率和科研效率。

#### ➤ 提升科研水平与质量

高质量的数据是科学研究的基础，平台能够确保数据的完整性和准确性，并提供数据分析工具和技术支持，帮助科研人员基于更全面、更准确的数据开展深入研究，从而推动地球化学领域的理论创新和技术进步。

#### ➤ 服务国家战略和社会需求

地球化学数据对于资源勘查、环境保护、气候变化研究等领域具有重要意义。

通过该平台，可以快速响应国家在矿产资源评估、环境污染治理、地质灾害预警等方面的战略需求，为决策制定提供科学依据。

#### ➤ 培育开放科学文化

建设地球化学数据管理平台符合国际上倡导的开放科学理念，有利于增强科研工作的透明度和可重复性，促进研究成果的社会效益最大化。

#### ➤ 强化国际合作与交流

平台有助于国内外科研团队之间的数据交换和合作研究，提高我国在全球地球化学研究领域的影响力和竞争力。

### 1.4 可行性分析

- **对现有系统的分析：**对于现有的基于网页的地质年代学实验室数据管理平台 Sparrow，在处理地球化学实验室数据方面可能存在的适用性差异。地球化学数据的特殊性包括不同的数据类型、测量方法和处理需求等。我们的系统将更好地支持地球化学实验室的需求。
- **经济可行性：**由于应用于实验室的数据库，硬件需求相对简单，因此成本主要集中在软件开发、人力资源、以及可能的培训方面。硬件成本相对较低，可以通过合理的选择满足需求的硬件设备来控制。整体成本估算应该在可接受范围内。
- **运维可行性：**地球化学数据管理平台的运维可行性得以提升，通过实施自动化部署、监控和警报系统、日志管理、容灾备份策略、自动化扩展以及安全性管理等措施。这使得系统在无人值守状态下稳定运行，降低了手动操作的需求，提高了可维护性、可靠性和稳定性。
- **安全可行性：**通过在用户本地部署并允许自主联网，地球化学数据管理平台可以根据用户的实际需求和安全标准进行灵活配置，从而提供更加安全可行的解决方案。

## 2. 系统关键技术

### 2.1 前端

#### ➤ 框架：Vue 2.6.14

Vue.js 是一种流行的 JavaScript 框架，用于构建用户界面和单页面应用程序。它的核心是一个响应式的数据绑定系统和一个灵活而强大的组件系统，使开发者能够构建可复用的、可维护的 web 应用程序。Vue 2.6.14 是 Vue.js 的一个特定版本，通常包含了一系列的改进和 bug 修复。

#### ➤ 组件库：Element-ui 2.15.13

Element-ui 是一个基于 Vue.js 的 UI 组件库，提供了一套丰富的、易于使用的组件，可以用来构建现代化的 Web 应用程序界面。这些组件包括按钮、表单、对话框、导航菜单等等，可以帮助开发者快速搭建漂亮且功能丰富的界面。版本号 2.15.13 表示这个组件库的特定版本，可能包含了一些新功能、改进和 bug 修复。

#### ➤ 开发工具：VSCode 1.82.2

Visual Studio Code（简称 VSCode）是一款由微软开发的轻量级、高性能的跨平台开源代码编辑器。它支持多种编程语言和文件格式，并提供了丰富的功能和扩展性，使开发者能够更高效地编写代码。版本号 1.82.2 表示这个编辑器的特定版本，可能包含了一些新功能、改进和 bug 修复。

### 2.2 后端

#### ➤ 语言 - Java8

Java 是一种广泛使用的高级编程语言，具有跨平台、面向对象、健壮性和安全性等特点。Java 8 是 Java 编程语言的一个版本，引入了许多新特性，其中包括函数式编程的支持、Lambda 表达式、流式 API、新的日期和时间 API 等。它为开发者提供了更多的编程工具和技术，使得 Java 编程更加简洁和高效。

#### ➤ 框架 - Springboot V2.5.5

Spring Boot 是一个用于简化 Spring 应用程序开发的框架，它基于 Spring 框架，通过自动配置和约定优于配置的原则，帮助开发者快速搭建和部署生产级别的应用程序。



Spring Boot v2.5.5 是 Spring Boot 框架的一个特定版本，通常包含了一系列的改进、新功能和 bug 修复，使开发者能够更轻松地构建出高质量的应用程序。

➤ **开发工具：Eclipse IDE 4.20.0**

Eclipse IDE 是一个功能强大的集成开发环境，用于编写、编译、调试和部署 Java 应用程序。它提供了丰富的功能，包括代码编辑器、调试器、版本控制集成、代码重构工具等，使得开发者能够更加高效地进行 Java 开发。版本号 4.20.0 表示这个 IDE 的特定版本，可能包含了一些新功能、改进和 bug 修复。

## 2.3 数据库

➤ **PostgreSQL 15.3**

PostgreSQL 是一个开源的关系型数据库管理系统，具有高度的稳定性、可靠性和扩展性。它支持大部分 SQL 标准，并提供了许多高级功能，如复杂查询、事务、触发器、视图等。PostgreSQL 15.3 是 PostgreSQL 数据库管理系统的一个特定版本，通常包含了一系列的改进、新功能和 bug 修复，以提高数据库的性能和可靠性。

➤ **相关工具：Navicat 16**

Navicat 是一个强大的数据库管理工具，提供了对多种数据库系统的支持，包括 PostgreSQL、MySQL、SQLite、Oracle 等。它具有直观的用户界面和丰富的功能，包括数据查询、数据编辑、数据同步、数据备份等。Navicat 16 是 Navicat 系列软件的一个特定版本，可能包含了一些新功能、改进和 bug 修复，以提供更好的数据库管理体验。

## 2.4 设计和管理工具

➤ **代码存储与协作：Github**



GitHub 是一个基于 Git 版本控制系统的代码托管平台，旨在帮助开发者存储、管理和协作开发代码。它提供了代码托管、问题追踪、代码审查、Wiki 等功能，使得团队能够更加轻松地共同开发和维护代码。GitHub 也被广泛用于开源项目的托管和贡献。

➤ **原型设计软件：Pixso 1.32.0.0**

Pixso 是一款用于创建交互式原型的设计软件，它提供了丰富的工具和组件，帮助用户

快速绘制和演示应用程序的用户界面。通过 **Pixso**，设计师和开发者可以创建高保真度的原型，进行用户测试和反馈，并与团队成员共享设计想法。

#### ➤ 接口文档管理工具：**ApiPost 7.2.1**

**ApiPost** 是一个用于管理和分享 API 接口文档的工具，它提供了简单直观的界面，可以帮助开发者轻松地编写、组织和共享 API 文档。通过 **ApiPost**，团队成员可以快速了解和使用 API，同时也可以方便地更新和维护接口文档。

## 3. 系统需求分析

### 3.1 需求定义

#### ➤ 需求概述

本系统的目标是建立一个全面的数据管理平台，旨在满足管理员和用户的多样化需求。在管理员功能方面，系统提供了安全的登录机制，以确保权限的合法性。同时，管理员可进行灵活的用户管理，包括注册审核和权限配置，以及数据审核功能，以保障上传数据的质量和合法性。系统还具备下载量统计功能，为管理员提供对数据使用情况的全面洞察。

对于用户而言，系统提供了用户友好的注册和登录流程，确保账户的安全性。用户可以使用系统的数据上传功能，支持多种格式，并能够自动进行格式验证。强大的数据检索功能允许用户进行多条件查询，而个人中心则方便用户管理个人信息和上传的数据文件。同时，查询历史记录功能有助于用户追溯和重复使用查询条件。

系统的共享功能包括模板下载，提供数据上传模板，以确保数据格式的一致性。此外，数据看板以可视化方式展示系统整体运行状况，包括用户活跃度和数据使用情况等信息。总体而言，系统设计旨在搭建一个高效、安全、用户友好的数据管理平台，同时满足管理员监控和审核数据的需求，为用户提供便捷的数据上传、检索和管理体验。

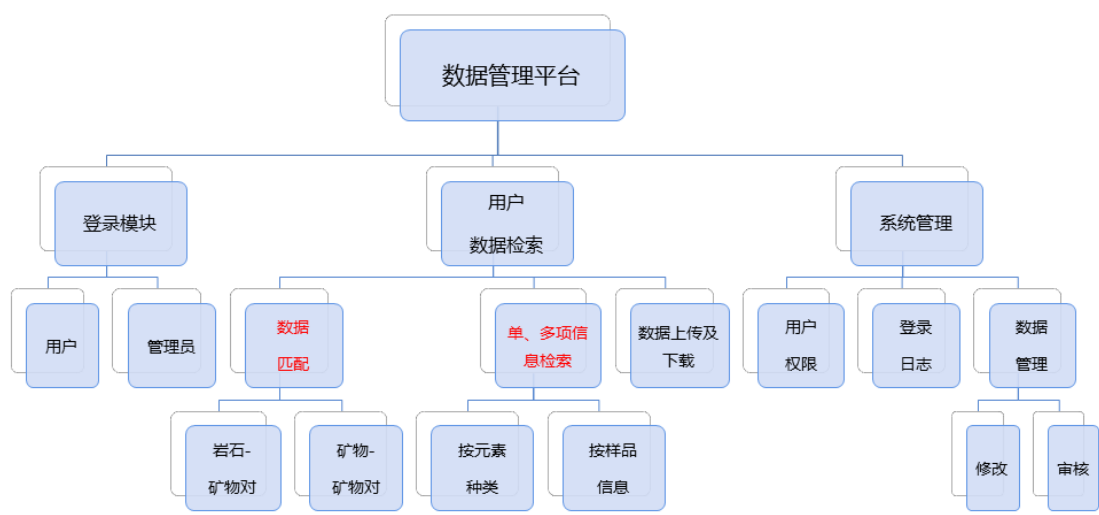
基于以上需求，我们设计了一个三层架构的地球化学实验室数据管理平台，包括数据存储层、数据处理层和用户界面层。具体架构如下：

**数据存储层：**数据存储层负责安全、可靠地存储和管理实验数据。我们采用了分布式数据库技术，将数据存储多个节点上，以提高数据的可用性和容错性。同时，我们

还采用了数据加密和访问控制等安全机制，以确保数据的安全性和隐私性。

数据处理层：数据处理层负责对实验数据进行处理和分析。我们采用了分布式计算技术，将数据处理任务分发到多个计算节点上，并利用并行计算和分布式存储等技术，实现对海量数据的高效处理和分析。同时，我们还提供了丰富的数据处理和分析工具，包括统计分析、机器学习、人工智能等，以满足不同研究需求。

用户界面层：用户界面层负责为用户提供友好、直观的操作界面。我们设计了一个基于 Web 的用户界面，可以在任何设备上通过浏览器访问，无需安装任何额外的软件。用户可以通过界面实现数据的浏览、查询、编辑、分析和可视化，并与其他用户进行实时的交流和协作。



➤ 需求规定

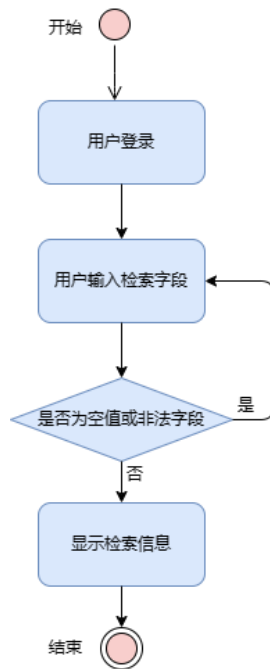
功能编号	所属父编号	功能定义	功能描述	优先级
1	无	用户功能集合	用户业务模块	5
2	无	管理员功能集合	管理员业务模块	5
3	2	管理员基本功能需求	管理员登录	4
4	1	用户基本功能需求	用户注册	4
5	1	用户基本功能需求	用户登录	4
6	1	用户基本功能需求	数据上传	3
7	1	用户基本功能需求	数据检索	3
8	2	管理员基本功能需求	数据审核	3
9	2	管理员基本功能需求	下载量统计	3

10	1	用户基本功能需求	用户个人中心	2
11	1	用户基本功能需求	查询历史	2
12	1	用户基本功能需求	模板下载	1
13	2	管理员基本功能需求	网站数据看板	1

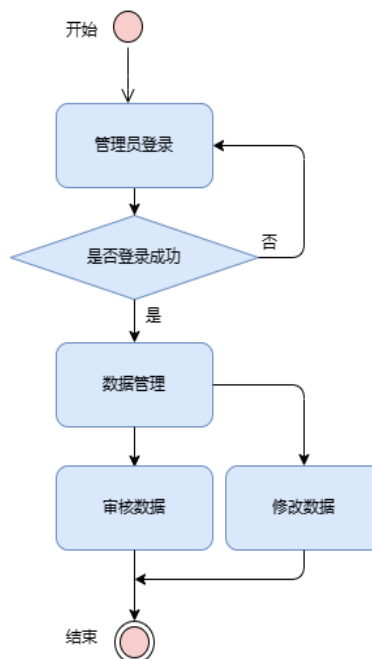
功能编号	详细功能描述
1	用户进入系统页面，能查询、下载和上传所需要的地球化学数据。
2	管理员进入系统页面，对用户上传的数据进行查看、审核及修改。
3	管理员在登录页面输入管理员账号信息登录系统
4	用户在注册页面输入基本个人信息完成注册
5	用户在登录页面输入用户账号信息登录系统
6	用户在上传数据页面导入数据文件，并填写样品类型等相关信息，最后点击上传按钮完成数据的上传。
7	用户输入岩石或矿物信息检索数据。
8	管理员在数据审核的页面对用户上传的数据进行审核和修改。
9	管理员在下载量统计页面查看各类数据的下载量。
10	用户在个人中心页面可以对个人信息进行查看和修改。
11	用户可以查询检索、上传数据和下载数据的记录。
12	用户可以下载模板来填写地球化学数据。
13	管理员可以在数据看板上对本网站浏览量、下载量等数据进行监测。

### 3.2 系统流程分析

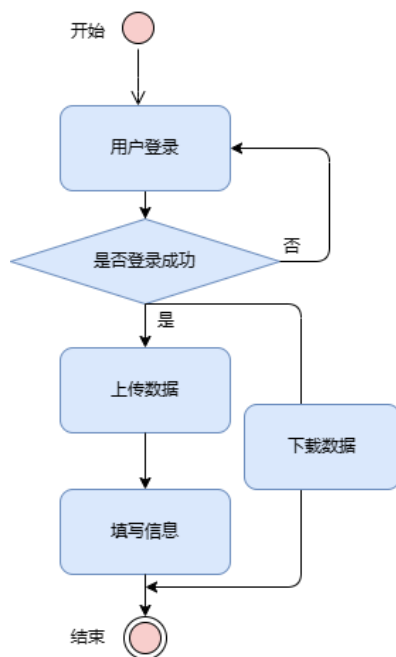
- **检索流程：**检索功能包括对岩石或矿物名称、样品地球年代等基本属性的检索。当输入的检索字段为空值或者包括非法字段时，则需重新输入检索信息。



- **数据管理流程:** 数据管理包括对数据的修改和审核。当数据满足所有预设条件并达到高质量标准，则该数据将顺利通过审核流程，并被安全、准确地导入到数据库系统中，以供后续查询、分析。当管理员发现有误数据时，管理员可以直接对数据进行修改，然后再上传至数据库系统。



- **数据上传和下载流程:** 用户登录完成后,可以在数据上传页面导入数据文件、填写相关信息上传地球化学数据; 也可以在数据下载页面下载所需的数据。



### 3.3 应用环境

- **硬件环境:**

**服务器:**

CPU: 云服务器常见配置, 或一般的工作站即可

内存: 16GB 及以上

硬盘: 根据课题组实际需求, 推荐 1TB 及以上

**管理员:**

设备: 任意 PC

**用户:**

设备: 任意 PC

- **软件环境:**

**服务器:**

数据库: PostgreSQL 14

后端: Springboot 2

前端: Vue 2.6

前端组件库: Element-UI 2

**管理员:**

Web 浏览器

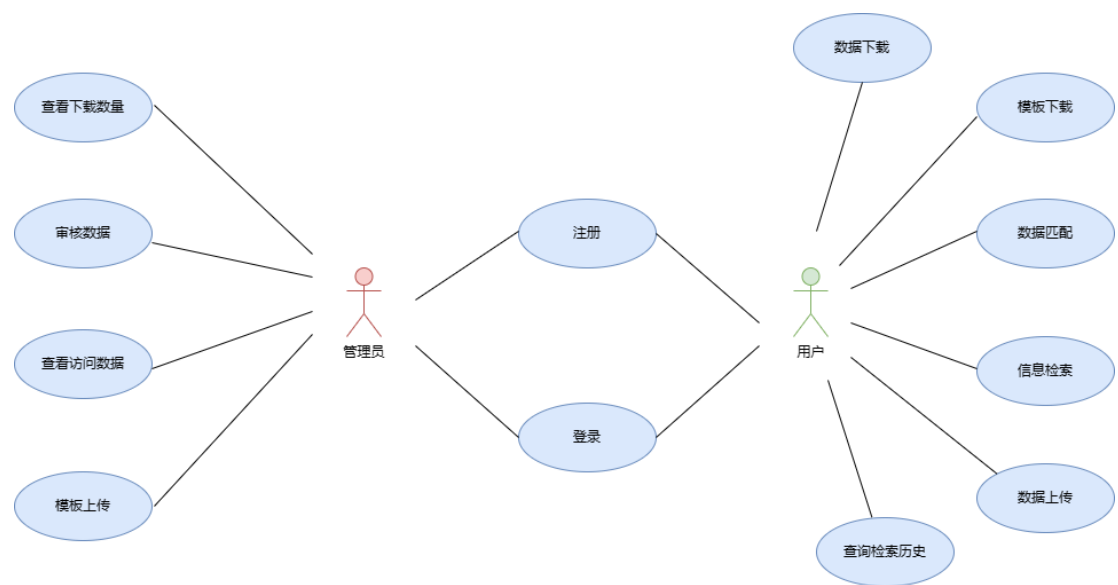
Navicat

**用户:**

Web 浏览器

## 4. 系统用例分析

### 4.1 系统用例图



### 4.2 系统用例文档

用例名	用户注册
简要描述	用户在系统中创建新的账号
参与者	用户
涉众	用户、系统管理员
相关用例	用户登录
前置条件	用户需要访问注册页面
后置条件	用户成功注册账号并可以使用该账号登录系统
基本事件流	
(1) 用户打开注册页面。	

<div><div>(2) 用户填写注册表格，包括用户名、密码、电子邮箱等必要信息。</div><div>(3) 用户点击注册按钮。</div><div>(4) 系统验证用户填写的信息是否符合规范，例如用户名是否已被使用、密码强度是否足够、邮箱格式是否正确等。</div><div>(5) 如果验证通过，系统将用户信息保存到数据库中，并显示注册成功的消息。</div><div>(6) 用户使用注册时填写的用户名和密码登录系统，验证注册是否成功。</div></div>
<div><div><div>备选事件流</div><div><div>A-1 用户名已存在</div><div><div>(1) 系统提示用户用户名已经被使用。</div><div>(2) 用户修改用户名，然后重新提交注册表格。</div><div>(3) 系统重新验证用户填写的信息。</div></div><div>A-2 邮箱格式不正确</div><div><div>(1) 系统提示用户邮箱格式不正确。</div><div>(2) 用户修改邮箱地址，然后重新提交注册表格。</div><div>(3) 系统重新验证用户填写的信息。</div></div><div>A-3 密码强度不足</div><div><div>(1) 系统提示用户密码强度不足，要求密码至少包含一定长度的字符并且包含数字、字母和特殊字符。</div><div>(2) 用户修改密码，然后重新提交注册表格。</div><div>(3) 系统重新验证用户填写的信息。</div></div></div></div></div>

用例名	用户登录
简要描述	用户使用已注册的账号登录系统
参与者	用户
涉众	用户、系统管理员



相关用例	用户注册、密码重置
前置条件	用户必须已经注册过账号
后置条件	用户成功登录系统，进入个人账号页面或应用主页
<b>基本事件流</b>  (1) 用户打开登录页面。 (2) 用户输入已注册的用户名和密码。 (3) 用户点击登录按钮。 (4) 系统验证用户输入的用户名和密码是否匹配已注册的账号信息。 (5) 如果验证通过，系统显示登录成功的消息，并跳转到个人账号页面或应用主页。 (6) 如果验证不通过，系统显示登录失败的消息，提示用户重新输入用户名和密码。	
<b>备选事件流</b>  <b>A-1 用户名忘记密码</b>  (1) 用户点击“忘记密码”链接。 (2) 系统跳转至密码重置页面。 (3) 用户输入注册时使用的邮箱地址或手机号码。 (4) 系统向用户提供重置密码的方式，例如通过邮箱发送重置链接或短信验证码。 (5) 用户根据系统提供的方式重置密码，然后返回登录页面重新登录。	

用例名	检索
简要描述	输入岩石或矿物相关的信息筛选相关的数据
参与者	用户
涉众	用户、管理员

相关用例	岩石或矿物相关信息的查询
前置条件	用户完成系统的登录
后置条件	用户可查看检索结果并能对检索结果进行下载及数据匹配
<b>基本事件流</b>	
<div>(1) 用户在系统中找到检索功能入口，并点击进入检索页面。</div> <div>(2) 用户输入合法的检索信息，包括岩石或矿物名称、类型、形成时间、形成条件、取样位置等，并点击检索按钮。</div> <div>(3) 系统根据用户输入的信息进行检索，并将符合条件的数据显示在页面上。</div> <div>(4) 页面显示检索结果，并提供数据下载和数据匹配的按钮。</div> <div>(5) 用户可以选择下载数据或进行数据匹配操作。</div>	
<b>备选事件流</b>	
<b>A-1 输入的检索信息非法</b>	
<div>(1) 系统检测到用户输入的检索信息非法，例如包含非法字符或格式不正确。</div> <div>(2) 系统显示空白结果，并提示用户输入的检索信息非法。</div> <div>(3) 用户需点击返回上一页的按钮，返回到检索页面，重新输入合法的检索信息进行查询。</div>	

用例名	上传
简要描述	用户选择要上传的数据文件，然后填写相关信息完成对数据上传。
参与者	用户
涉众	用户、管理员
相关用例	下载
前置条件	用户完成对系统的登录并点击进入数据上传页面

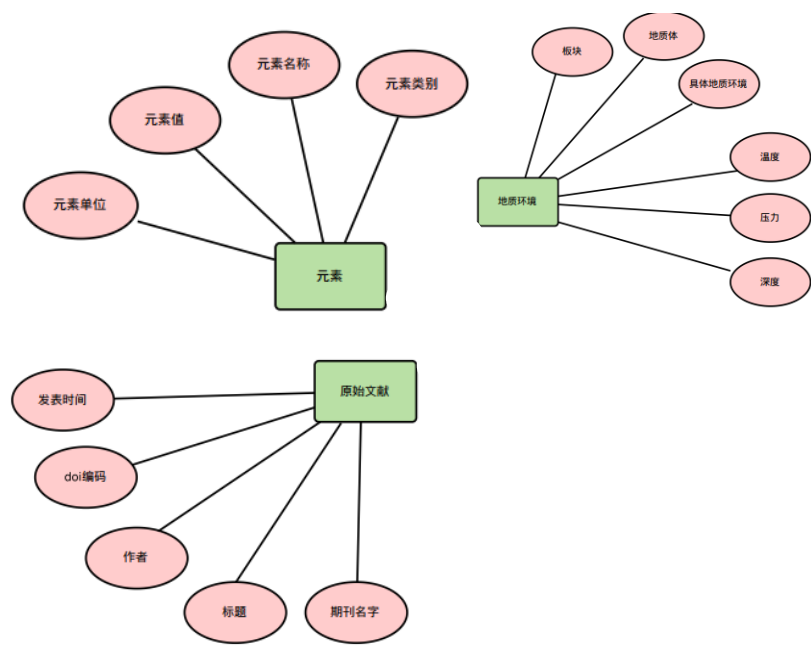
后置条件	数据成功上传至数据库系统
<b>基本事件流</b>	
<div><div>(1) 用户在系统中找到数据上传功能入口，并点击进入数据上传页面。</div><div>(2) 用户点击上传文件按钮，系统弹出文件选择对话框。</div><div>(3) 用户从计算机中选择要上传的数据文件，并确认选择。</div><div>(4) 系统接受文件上传并显示文件名。</div><div>(5) 用户填写样品的相关信息，包括样品名称、采样日期、采样地点等。</div><div>(6) 用户填写样品采样者的相关信息，包括采样者姓名、单位等。</div><div>(7) 用户点击上传按钮。</div><div>(8) 系统接收上传的数据文件和填写的信息，将数据存入数据库，并显示上传成功的消息。</div></div>	
<b>备选事件流</b>	
<b>A-1 文件上传失败</b>	
<div><div>(1) 系统检测到文件上传失败，可能是由于网络问题或文件格式不支持。</div><div>(2) 系统提示用户文件上传失败，并提供重新选择文件的选项。</div><div>(3) 用户重新选择文件，然后再次尝试上传。</div></div>	
<b>A-2 信息空缺或填写有误</b>	
<div><div>(1) 系统检测到用户填写的信息中有空缺或填写有误。</div><div>(2) 系统提示用户填写或重新填写信息，并标明哪些信息需要修正。</div><div>(3) 用户根据系统提示填写或修改信息。</div><div>(4) 用户点击上传按钮，再次尝试上传数据。</div></div>	

用例名	下载
简要描述	用户对检索结果进行下载
参与者	用户
涉众	用户：用户在检索结果的页面上点击下载按钮并选择路径

	即可对检索的数据进行下载
相关用例	页面上文件的下载
前置条件	检索页面显示检索结果
后置条件	提示下载完成
<b>基本事件流</b>  (1) 用户在检索结果页面找到要下载的数据项。 (2) 用户点击下载按钮。 (3) 系统弹出文件下载对话框，并提示用户选择下载路径。 (4) 用户选择下载路径，并确认下载。 (5) 系统开始下载数据文件。 (6) 下载完成后，系统提示用户下载完成。	
<b>备选事件流</b>  <b>A-1 下载失败</b>  (1) 下载过程中出现网络故障或其他问题，导致下载失败。 (2) 系统提示下载失败，并提供重新下载的选项。 (3) 用户重新点击下载按钮，并选择合适的路径进行下载	

# 5. 数据库

## 5.1 数据库关系图



## 5.2 数据库核心结构 ER 图



### 5.3 数据库结构解释

➤ 不使用外键

在数据库设计中，使用外键是确保数据完整性和一致性的一种常见方式。然而，有时候不使用外键可能更有利。不使用外键的情况包括：改、删时需要考虑外键约束，增加了操作的复杂性和风险；外键约束可能导致表级锁，影响并发性能；外键的级联删除特性可能导致数据不可控，例如删除主表记录会导致相关从表记录被删除；使用外键增加了表与表之间的耦合度，迁移数据库结构时需要先解除外键约束，可能导致数据丢失或需要更多的操作步骤。因此，在一些情况下，如需要更高的并发性能、更灵活的数据操作或简化数据库结构时，选择不使用外键可能是一个合理的选择。然而，这也需要在设计时权衡利弊，并根据具体业务需求做出决策。

### 5.4 数据库字典

序号	字段名称	类型	中文释义
1	samp_id	int4	样品编号
2	samp_age	numeric(20, 10)	样品年龄
3	exper_id	int4	实验样品编号
4	test_method	varchar(255)	年龄检测方法
5	origin_num	varchar(255)	样品原始编号
6	rock_id	int4	岩石样品编号
7	mineral_id	int4	矿物样品编号
8	loc_id	int4	位置编号
9	environ_id	int4	地质环境编号
10	parent_id	int4	父样品编号
11	collect_id	int4	样品收集编号
12	age_error	numeric(20, 10)	年龄误差
13	refer_id	int4	文献编号
14	Notes	varchar(255)	补充说明
15	samp_type	varchar(50)	样品类型
16	samp_weight	numeric(20, 10)	样品重量
17	samp_volume	numeric(20, 10)	样品体积
18	sample_date	date	采样日期
19	storage_location	varchar(255)	存储位置
20	collector_id	int4	采样人员编号
21	collection_notes	text	采集备注
22	analysis_date	date	分析日期

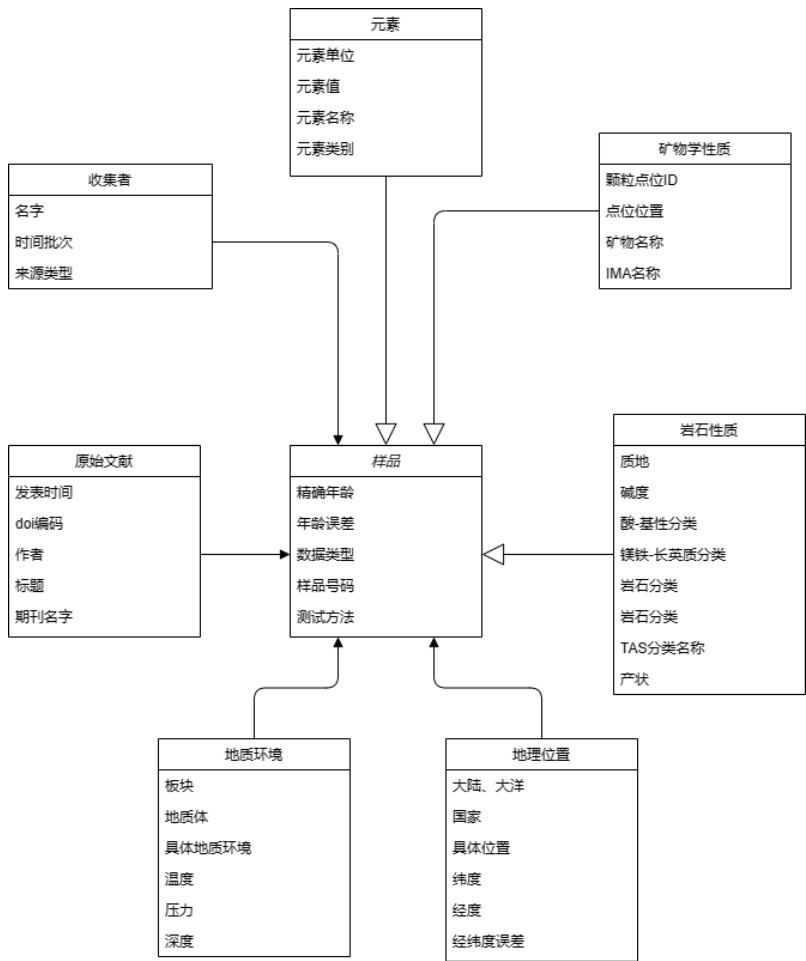
23	analyst_id	int4	分析人员编号
24	analysis_notes	text	分析备注
25	created_at	timestamp	创建时间
26	updated_at	timestamp	更新时间
27	exper_date	date	实验日期
28	experiment_notes	text	实验备注
29	test_result	varchar(100)	测试结果
30	result_unit	varchar(50)	结果单位
31	result_error	numeric(20, 10)	结果误差
32	result_notes	text	结果备注
33	reference_notes	text	参考文献备注
34	analysis_method	varchar(255)	分析方法
35	location_name	varchar(100)	位置名称
36	location_coordinates	varchar(50)	位置坐标
37	environment_description	text	环境描述
38	parent_samp_type	varchar(50)	父样品类型
39	parent_samp_origin	varchar(255)	父样品来源
40	parent_samp_age	numeric(20, 10)	父样品年龄
41	analysis_result	text	分析结果
42	analysis_summary	text	分析摘要
43	experiment_conditions	text	实验条件
44	test_conditions	text	测试条件
45	lab_location	varchar(100)	实验室位置
46	equipment_used	varchar(255)	使用设备
47	experiment_duration	interval	实验持续时间
48	analysis_duration	interval	分析持续时间
49	sample_preparation_notes	text	样品制备备注
50	analysis_quality_control	text	分析质量控制

## 6. 系统架构

### 6.1 架构详细



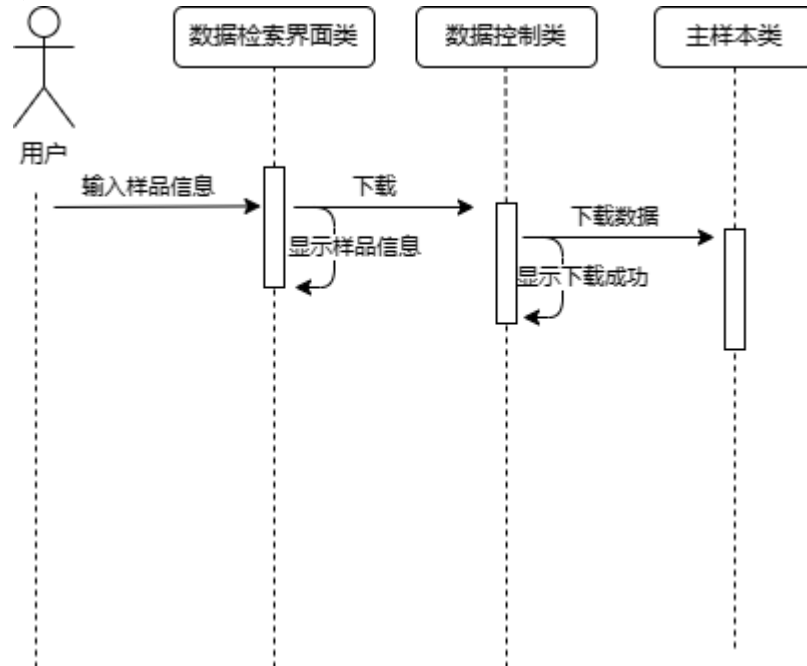
6.2 系统类图



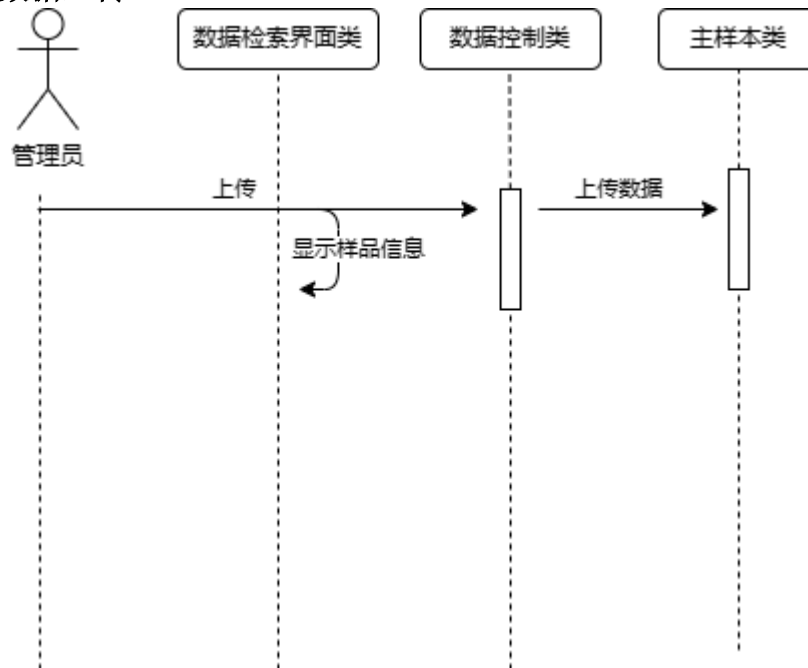


## 6.3 系统详细设计

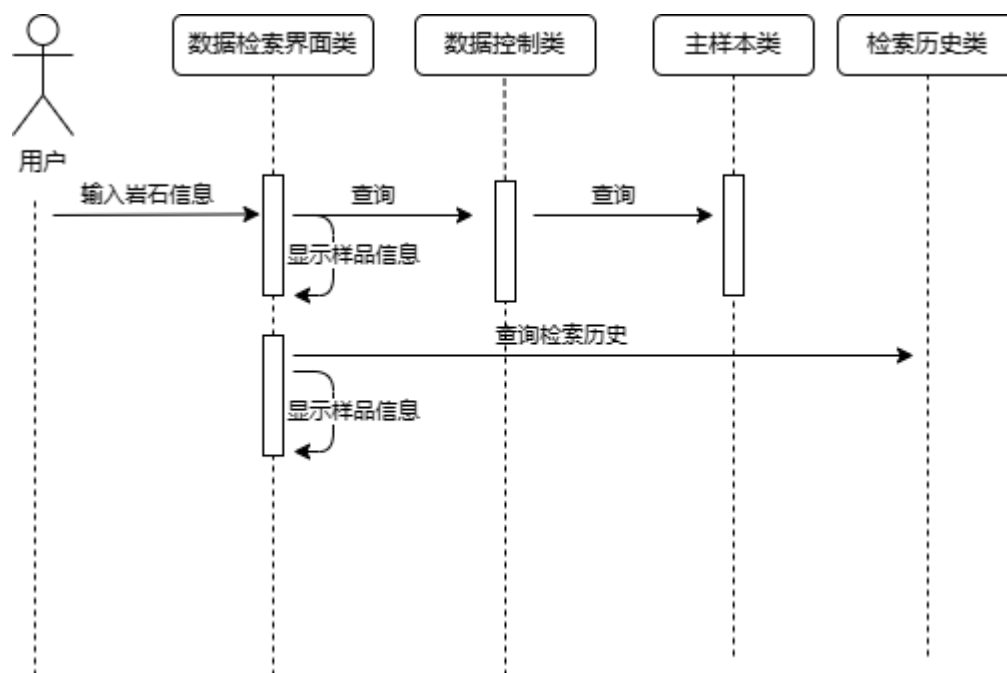
### ➤ 数据下载:



### ➤ 数据上传:



### ➤ 数据检索:



## 7. 系统展示和说明

### 7.1 原则

#### ➤ 用户界面设计原则：

（1）清晰性与简洁性：界面布局直观明了，避免杂乱无章的信息堆砌，突出关键功能和数据视图。使用清晰、专业的术语和图标，确保用户能够快速理解各种元素的含义和功能。提供足够的空白空间和适当的分隔，增强视觉层次感，降低认知负担。

（2）响应式设计：交互元素应易于触控操作，并根据设备特性调整大小和间距。

（3）导航与搜索：采用清晰、一致的导航结构，使用户能够轻松浏览不同模块和数据集。提供强大的搜索功能，支持矿物类型、矿物年龄、地理位置查询等多种方式，快速定位所需信息。

#### ➤ 界面一致性

（1）风格统一：维持一致的颜色、字体、图标样式和交互模式，构建连贯的品牌形象和视觉语言。遵循平台自身的 UI 设计指南，确保新添加或更新的部分与已有界面无缝融合。

（2）布局一致性：各页面和组件的布局应遵循相同的设计模式和网格系统，确保用户在不同页面间切换时感到熟悉和舒适。对齐、间距、分组规则等视觉元素应保持一致，

增强界面的整洁性和可预测性。

(3) 交互一致性：功能相似的操作应有相同的触发方式和反馈效果。

### ➤ 用户帮助措施

(1) 内置文档与教程：提供详细的在线帮助文档，涵盖平台功能介绍、操作指南、数据格式说明等内容。

(2) 设置明显的“反馈”或“帮助”入口，方便用户报告问题、提出改进建议或寻求技术支持。

### ➤ 出错信息和警告

(1) 错误消息应准确描述问题所在，包括错误类型、涉及的数据或操作、可能的原因及解决方案建议。

(2) 错误信息应出现在用户视线焦点附近，确保用户不会错过。

(3) 提供明确的下一步行动指导，如重试、撤销操作、跳转到相关设置页面等。

### ➤ 一般性交互原则

(1) 反馈及时性：对用户的操作给予即时反馈，如加载动画、确认消息、状态变化提示等，让用户了解系统正在响应其请求。

(2) 操作可控性：提供撤销/重做功能，允许用户纠正误操作。

### ➤ 数据输入原则

(1) 数据校正：实施严格的前端和后端数据验证，确保输入数据符合预期格式、范围、唯一性等约束。对不符合规范的数据提供实时反馈，提示用户修正。

(2) 批量处理：支持批量导入、编辑、删除数据，提高数据管理效率。

(3) 输入辅助：对于专业词汇或编码系统，提供下拉列表、联想输入等辅助工具。提供自动完成功能，减少用户输入负担，提高数据录入速度。

## 1.2 用户界面

### ➤ 主界面

主界面功能丰富，提供了数据检索与匹配、参考的 Excel 表格、数据上传、联系方式、用户注册与登录等选项。同时，还实时展示了该平台数据库收集的岩石数量等相关数据，让用户能够轻松地获取所需信息。无论是查找特定数据还是上传自己的数据，都能在这个界面上完成，为用户提供了便利和高效的体验。

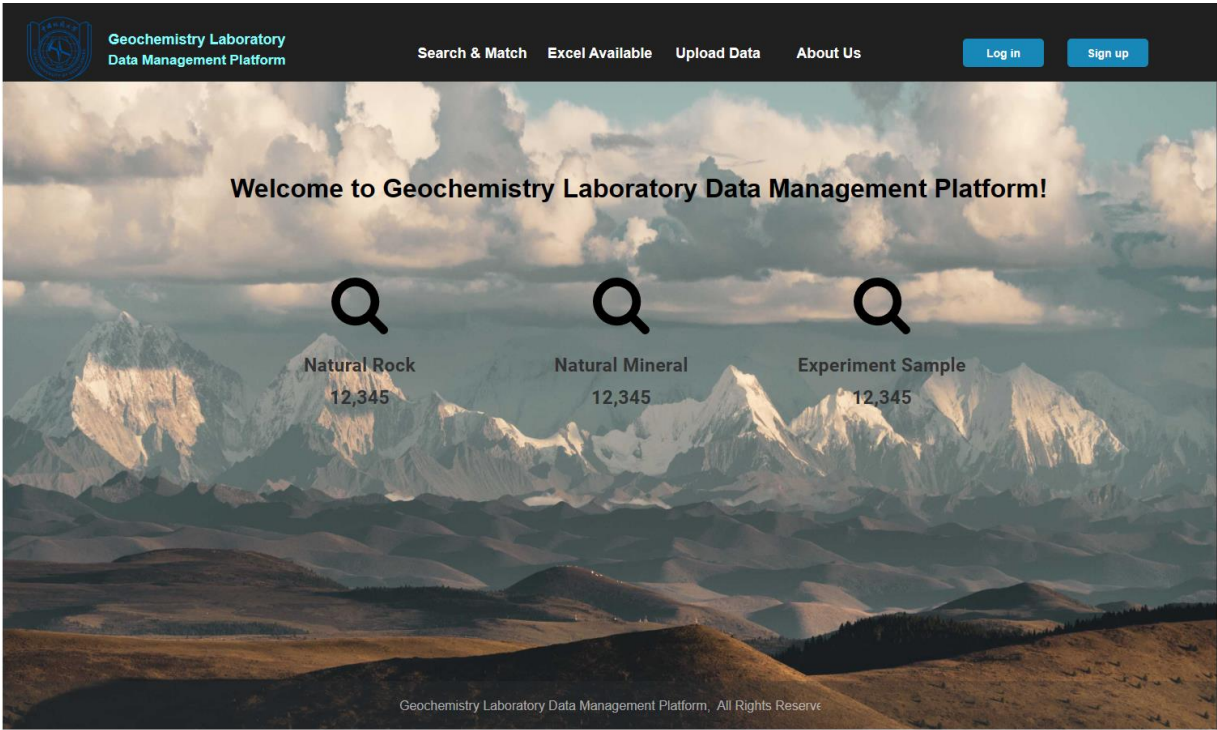


图 1 主界面

## ➤ 用户登录界面

该界面设计了用户友好的账号和密码输入框，让用户能够轻松输入登录信息。此外，还提供了可记住账号密码的选项，以方便用户下次登录时快速访问。登录按钮则使用户能够立即访问他们的个人账户。

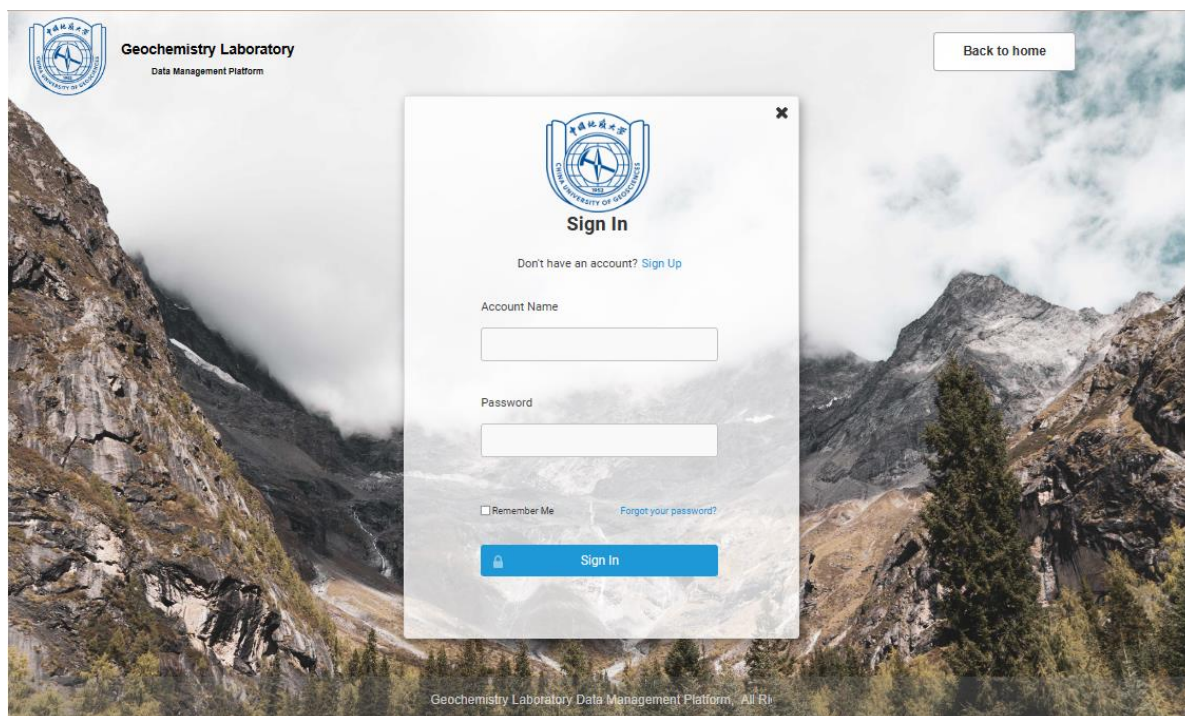



图 2 用户登录界面

➤ 检索界面

用户可通过输入岩石种类、年龄等六项相关数据进行数据检索，并为用户提供了输入辅助功能，用户可以通过下拉列表找到合适的检索信息进行高效检索。



Geochemistry Laboratory  
Data Management Platform

Search - natural rock

Back to home

Let's customize your rock data filter!

Rock Type

rock name

Choose

attitude

please select

alkalinity

please select

mafic/felsic

please select

clear

filter

Age

age (Ma)

minmax

chronology

Choose

test method

please select

clear

filter

Elements & Isotopes

Select Elements	Unit	Max	Min	
<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
<div>+ Add another</div>				

clear

filter

Location

continent/ocean

please select

country

Choose

longitude

minmax

latitude

minmax

clear

filter

Geological Setting

plate

please select

terrane

Choose

P (GPa)

minmax

T (°C)

minmax

depth (km)

minmax

clear

filter

Data Source

source type

please select

publish year

minmax

journal

Choose

first author

Choose

clear

filter

Note: The default(blank) option is to select all.

Clear all

Filter all

图 3 检索界面



## ➤ 查询结果界面

该页面以表格形式展示查询结果，并提供了下载数据、匹配数据等按钮，同时在界面的右下方展示了该用户账号的查询历史。

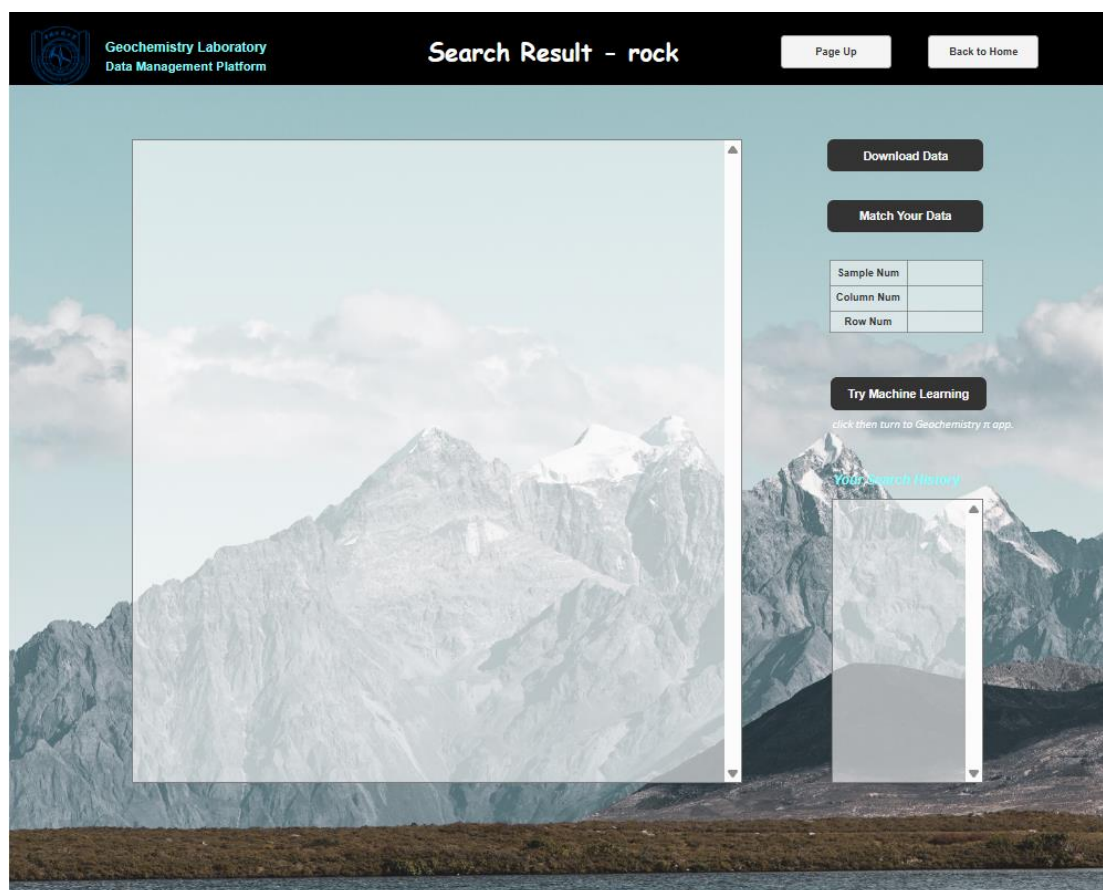


图 4 查询结果界面

### ➤ 1.2.5 数据上传界面

该界面被分为三个有序的子界面，用户按照上方长方形框的提示，依次完成信息输入即可轻松上传数据。第一个子界面提供了上传文件的按钮，用户可以方便地选择要上传的文件。第二个子界面包含样品信息输入框，用户可以填写与样品相关的信息。第三个子界面包含收集者信息的输入框，用户可以提供有关收集者的信息。这三个步骤让数据上传变得简单明了，确保用户能够顺利完成任务。

Geochemistry Laboratory  
Data Management Platform

## Upload Data Online

Back to Home

1 Step 1  
Choose data file

2 Step 2  
Fill data description

3 Step 3  
Submit data

Upload File

or drag the file here to upload it  
(xlsx, xls, csv)

Note: Please ensure that your data format fits the data template of this database.  
click here to [data template](#)

Cancel Next





1 Step 1

Choose data file

2 Step 2

Fill data description

3 Step 3

Submit data

Collector Name\*

Collector Email\*

Organization

Collect Time\*

 /  / 

Sample Type\*

Sample Name\*

Collect Location\*

Source Type\*

Note:

[Page Up](#)

[Next](#)



1 Step 1

Choose data file

2 Step 2

Fill data description

3 Step 3

Submit data

Data Path:

Collector Name\*

Collector Email\*

Organization

Collect Time\*

 /  / 

Sample Type\*

Sample Name\*

Collect Location\*

Source Type\*

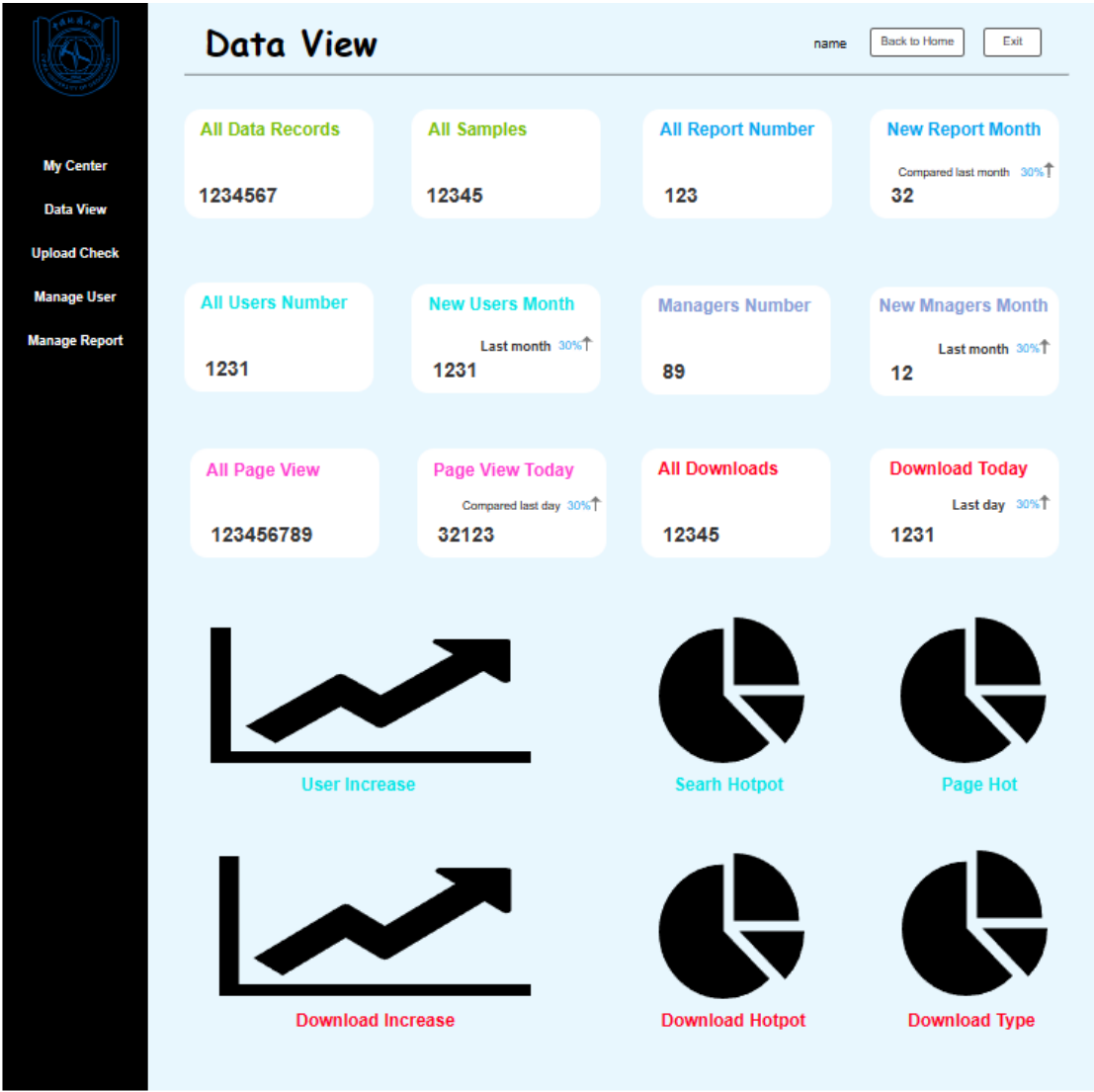
Note:

[Page Up](#)

[Submit](#)

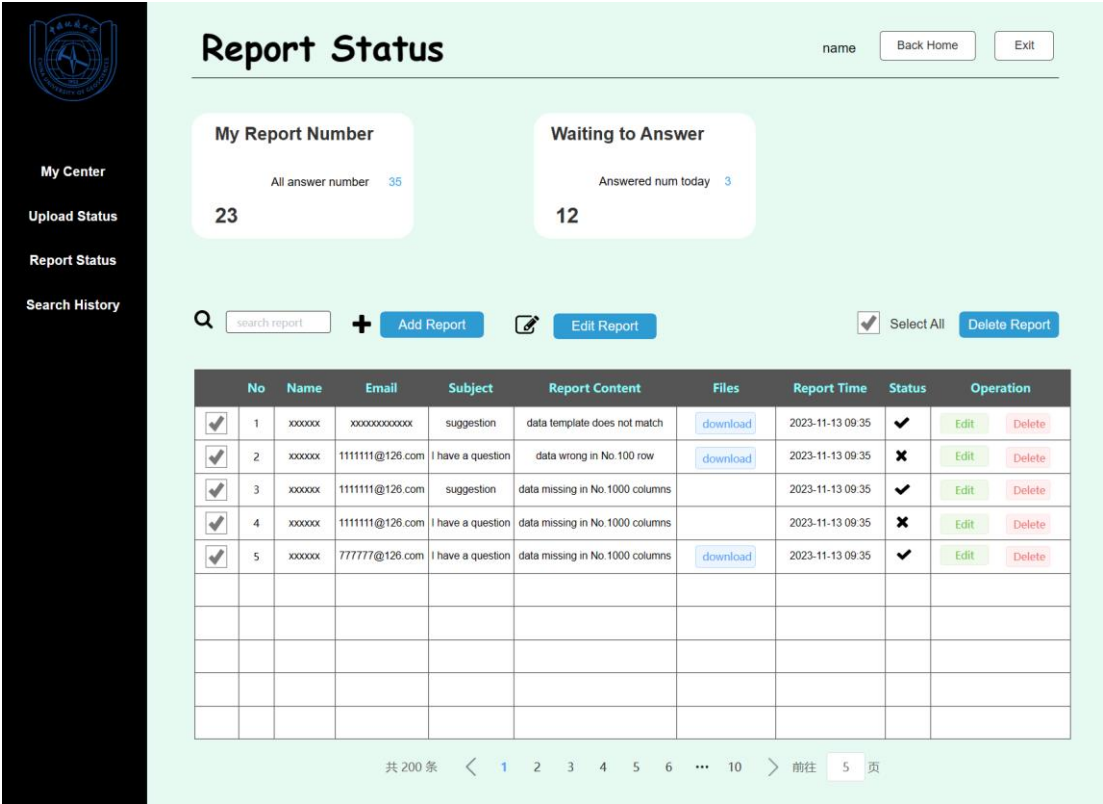
➤ 1.2.6 数据视图界面

该界面提供了对管理员实时展示的功能，包括用户数量、数据上传量、下载量、样本数量等平台记录。这些数据被直观地呈现，并进行了可视化处理，使管理员能够轻松地监测平台的实时情况。通过直观的图表和统计数据，管理员可以迅速了解平台的运行状态，及时做出相应的调整和决策，确保平台运行的顺利和高效。



➤ 1.2.7 反馈界面

该界面集中了所有用户提交的反馈信息，为管理员提供了一个方便的查看和管理平台用户意见的地方。通过这个功能，管理员可以及时了解用户的需求、建议和问题，以便针对性地改进平台功能和服务。这样的反馈机制不仅增强了用户参与感，也有助于提升平台的用户体验和满意度。



8. 白盒测试

8.1 注册功能测试

➤ 测试代码

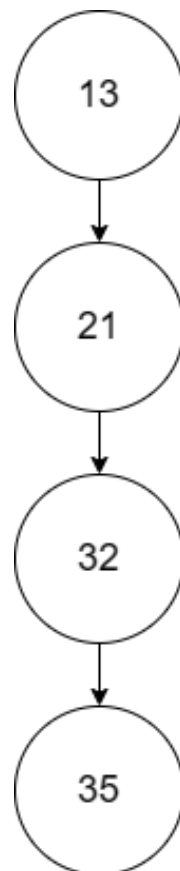
```
9. public class UserServiceTest {
10.
11.     private UserService userService;
12.     private UserRepository userRepository;
13.
14.     @Before
15.     public void setup() {
16.         userService = new UserService();
```

```

17.     userRepository = mock(UserRepository.class);
18.     userService.setUserRepository(userRepository);
19. }
20.
21. @Test
22. public void testRegister() {
23.     String username = "newuser";
24.     String password = "password123";
25.     User newUser = new User(username, password);
26.
27.     when(userRepository.save(newUser)).thenReturn(newUser);
28.
29.     boolean isRegistered = userService.register(newUser);
30.
31.     assertTrue(isRegistered);
32. }
33. }

```

➤ 注册控制流程图



➤ 输入输出表

测试用例序号	newuser	Password123	对应路径基本集
--------	---------	-------------	---------

1	用户名符合要求	密码符合要求	13,21,32,35
2	用户名不符合要求	*	13,21,32,35
3	用户名符合要求	密码不符合要求	13,21,32,35

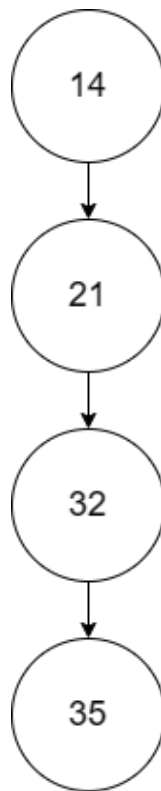
## 8.1 登录功能测试

```

1. public class UserServiceTest {
2.
3.     private UserService userService;
4.
5.     @Before
6.     public void setup() {
7.         userService = new UserService();
8.     }
9.
10.    @Test
11.    public void testLogin() {
12.        String username = "testuser";
13.        String password = "password123";
14.        User expectedUser = new User(username, password);
15.
16.        User loggedInUser = userService.login(username, password);
17.
18.        assertEquals(expectedUser, loggedInUser);
19.    }
20.}

```

➤ 登录控制流程图：



➤ 输入输出表

测试用例序号	testuser	Password123	对应路径基本集
1	数据库已存在用户名	用户名对应账号	14,21,32,35
2	数据库不存在用户名	*	14,21,32,35
3	数据库已存在用户名	错误密码	14,21,32,35

## 9. 黑盒测试

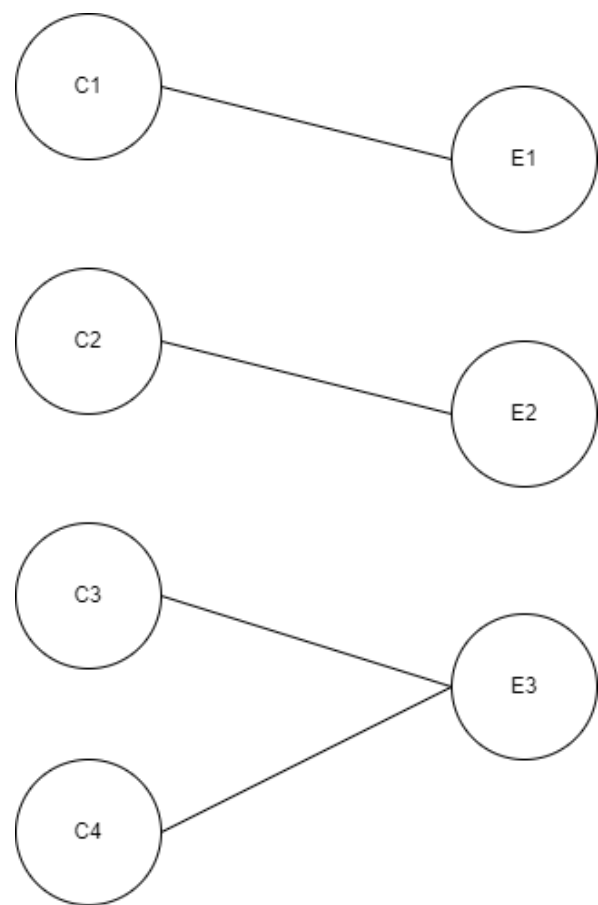
### 9.1 注册功能测试

➤ 输入输出表

原因	结果
C1 输入已存在的用户名注册。	E1 返回错误信息“用户名已存在”。
C2 输入无效的电子邮件地址注册。	E2 返回错误信息“电子邮件地址无效”。
C3 输入密码过短注册。	E3 返回错误信息“密码过短”。
C4 注册表单为空点击注册按钮。	E4 返回错误信息“所有必填字段不能

	为空”。
--	------

➤ 因果图



➤ 决策表

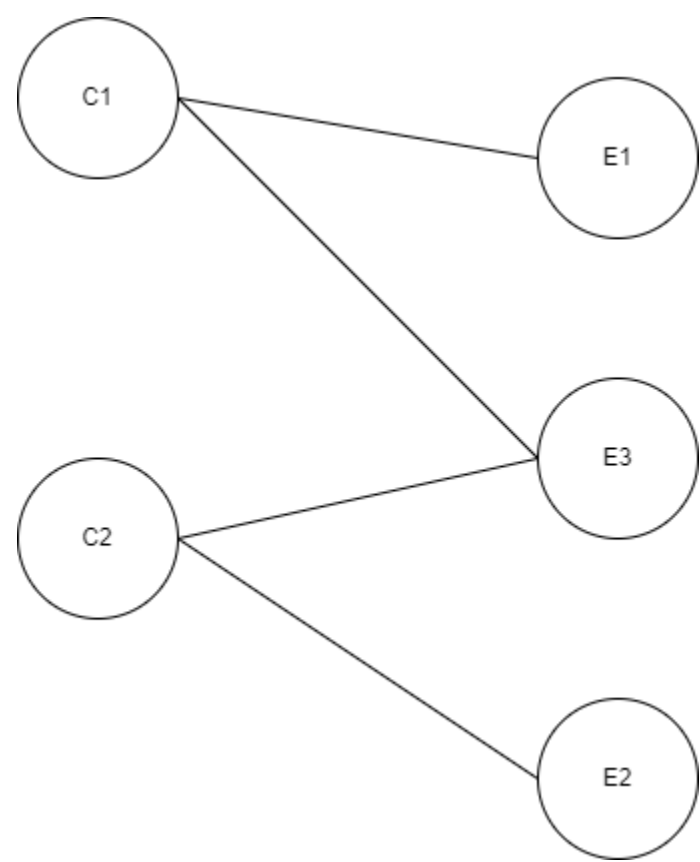
序号		1	2	3	4
条件	C1	1	0	0	0
	C2	0	1	0	0
	C3	0	0	1	0
	C4	0	0	0	1
动作	E1	1	0	0	0
	E2	0	1	0	0
	E3	0	0	1	0
	E4	0	0	0	1
用例		输入已存在的用户名注册	输入无效的电子邮件地址	输入密码过短	注册表单为空点击注册按钮

9.2 登录功能测试

➤ 输入输出表

原因	结果
C1 用户不存在	E1 返回错误信息“用户名不存在”
C2 密码错误	E2 返回错误信息“密码错误”
	E3 返回登录界面
	E4 登录成功

➤ 因果图



➤ 决策表

序号		1	2
条件	C1	1	0
	C2	0	1
动作	E1	1	0
	E2	0	1
	E3	1	1
	E4	0	0
用例		用户不存在	密码错误

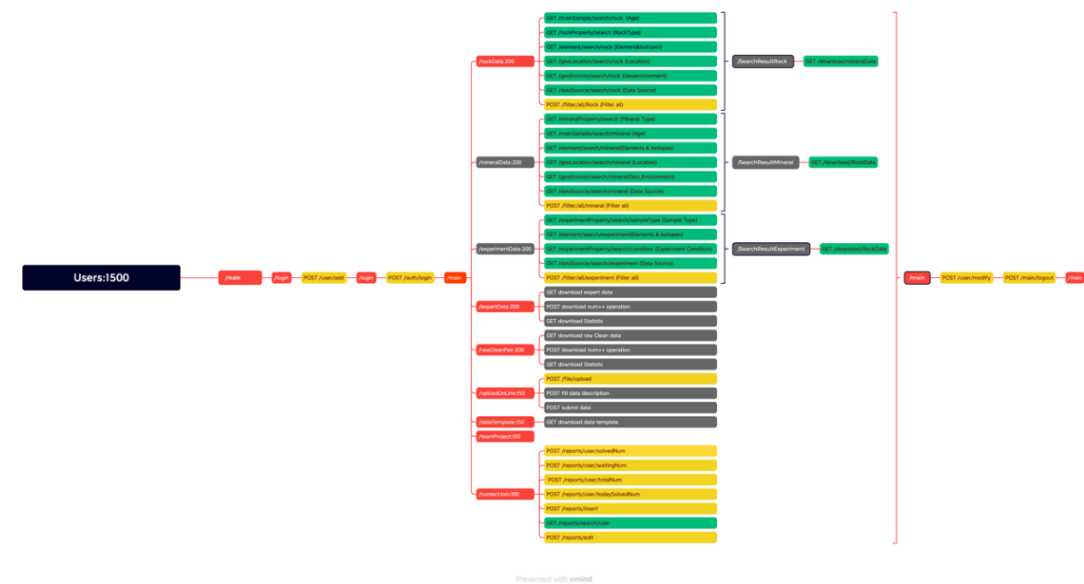


# 10. 压力测试

## 10.1 目标和指标

- 吞吐量（RPS）：每秒钟处理的请求数量
- 响应时间（RT）：从发出请求到收到响应的的时间。
- 错误率（failure）：请求失败或错误响应的比率。
- 并发用户数：同时在线的用户数量。

## 10.2 场景设计



## 10.3. 数据准备

- 数据负载
  - 准备适当的数据负载
  - 测试数据录入数据库

不同大小的文件随机上传

➤ **数据清洗**

确保测试数据的一致性和准确性。

## **10.4. 压力测试配置**

➤ **用户分布**

模拟不同地区、设备、网络条件下的用户分布

➤ **负载模式**

并发用户数（1500）、增长速率（2）、持续时间（24h）

➤ **请求类型**

不同类型的 HTTP 请求（GET、POST）以及页面访问

➤ **实时资源监控**

监控测试期间的各种指标和性能数据