项目开发文档

**一、项目概况**

1.题目：**大学软科排名可视化系统**

2.选题背景：

**1. 高等教育发展：**随着全球高等教育的快速发展，学校之间的竞争日益激烈。不同高校在科研、教学等方面的表现需要有效的评估和比较。

**2. 信息透明化需求：**学生和家长在选择大学时希望获得详尽的信息，帮助他们做出明智的选择。可视化系统能够将复杂的数据以直观的方式呈现，满足这一需求。

**3. 数据驱动决策：**教育政策制定者、学术机构及相关利益方希望通过数据分析来评估和改善教育质量。可视化工具可以帮助他们更好地理解和利用数据。

**4. 多维度评估：**传统排名往往侧重于单一指标，而现代教育评价需要考虑多种因素，如教学质量、科研能力、师资力量等。可视化系统能够整合多个维度的数据，提供全面的分析。

**5. 国际化趋势：**随着高等教育的国际化，学生和学者越来越关注国际排名。建立一个有效的可视化系统，可以帮助用户在全球范围内进行比较与分析。

**二、需求分析**

1.功能需求分析

**大学软科排名可视化系统**

**用户界面**

数据展示

数据查询

**后端服务**

**数据库**

数据存储

数据处理

数据查询

API接口

**用户界面：**

**数据展示：**排名图表，使用条形图、地图、饼图等图表展示关于大学软科排名的数据变换，便于用户直观比较。这些图表直观的反映了各省份在高等教育领域的竞争力与展示和比较不同大学和地区的教育特点，帮助人们更好地了解中国高等教育的分布和水平，更好地了解各地区的发展状况和教育特点

**数据查询：**使用 Vue.js 框架结合 Axios 库来发起 HTTP 请求，调用后端 API。这个过程中，前端接收后端返回的 JSON 格式数据。接着，前端 JavaScript 代码解析这些数据，使用了ECharts库来渲染图表，并将其展示在网页上。

**后端服务：**

**数据处理：**数据爬取，使用Python的爬虫工具（如Scrapy或Beautiful Soup）从公开数据源抓取高校排名信息。定期调度爬虫任务，确保数据的时效性。

**API接口：**数据获取API，写入数据与读取数据。

**数据库：**

**数据存储：**数据更存储在分布式数据库hbase中，利用HBase存储和处理大规模的教育数据，结合可视化技术，系统能够揭示教育资源在不同地区、不同类型大学之间的分布情况。这有助于教育管理者发现资源配置的不均衡，优化教育资源的分配，推动教育公平。

**数据查询：**基本查询，模糊搜索，多条件查询，对比查询，统计查询。

2.非功能需求分析

**扩展性**

1. 模块化设计：

采用模块化架构，便于在未来添加新功能，例如支持更多排名指标或数据源。

2. 数据接口：

设计灵活的数据接口，允许与外部数据库或API集成，使得数据更新和扩展更为简单。

3. 多平台支持：

支持不同平台（如Web、移动端等）的访问，确保用户可以在各种设备上使用系统。

4. 前端框架的灵活性：

使用现代前端框架（如Echart、Vue.js等）来构建用户界面，确保界面能够随着功能扩展而快速迭代。

**稳定性**

1. 错误处理机制：

实现全面的错误处理和日志记录机制，以快速识别和解决潜在问题，提高系统的稳定性。

2. 数据库管理：

使用可靠的数据库管理系统，定期备份数据，确保数据完整性和安全性，从而提高系统的稳定性。

3. 测试和验证：

定期进行系统测试（单元测试、集成测试等），确保各个模块之间的兼容性和整体系统的稳定性。

**三、项目设计**

1. 主界面：

大学软科排名可视化系统

大学类别的占比

各省大学数量

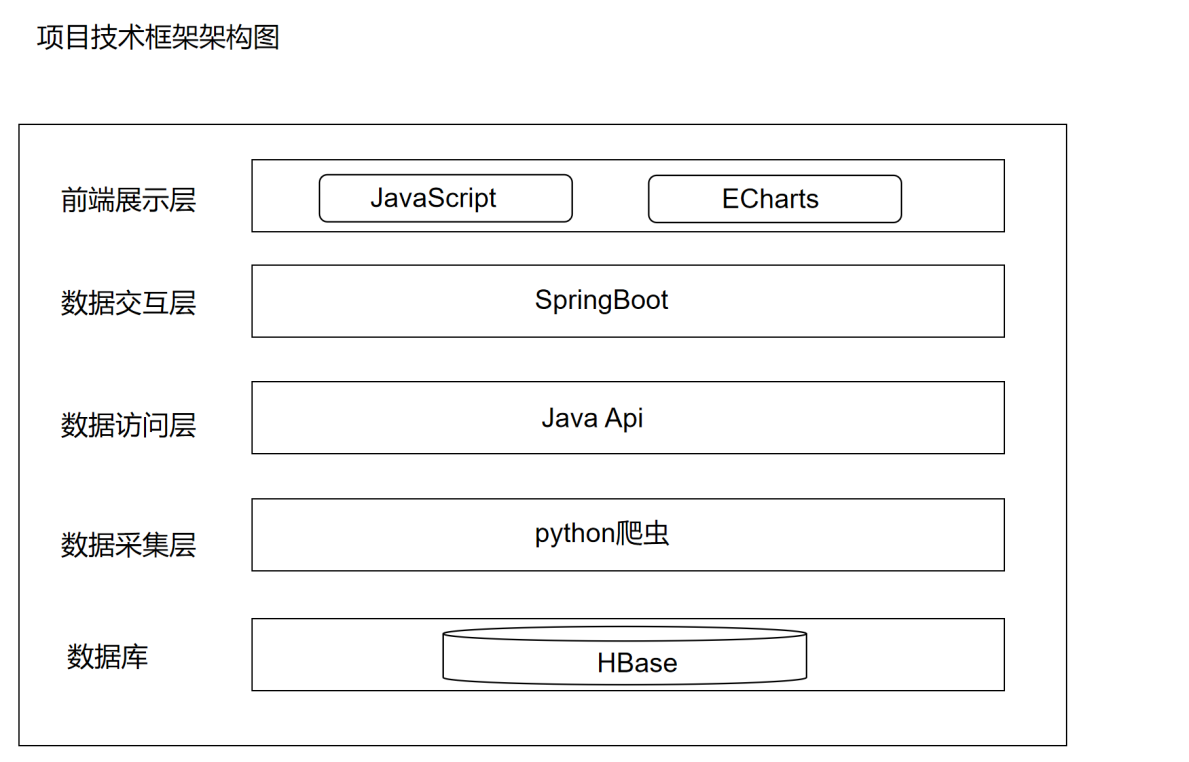
各省大学列别数量

各地区大学类型分布

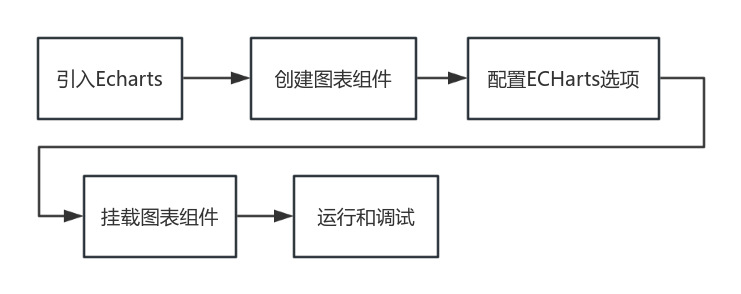
各大学的办学水平

大学软科平均总分

1. 项目框架

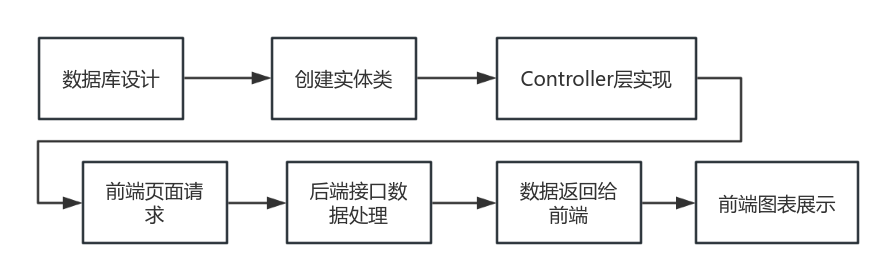


1. 功能设计
2. 网页功能：使用echart+vue实现网页界面数据图表的展示。

流程图：

算法设计：接受后端发送而来的数据，对数据进行处理，再通过echart图表来展示。

1. 后端功能：使用idea和pycharm来编写代码。

流程图：

算法设计：使用py文件进行爬取，idea进行网页框架的搭建。

3.数据库设计

**使用hbase数据库：**

**数据表结构设计：**

数据表名称：大学软科排名。

列名：大学名称，省份名称，大学类别，大学分布的地区，软科平均总分。

**数据的更新和储存：**

数据更新：HBase 支持快速的数据插入和更新操作，使用 put 方法可以轻松更新数据。

性能优化：预分区：在创建表时可以进行预分区，以提高写入性能。使用 BulkLoad：对于大批量数据插入，可以使用 HBase 的 BulkLoad 功能，减少写入延迟。合理设计 Row Key：选择合适的 Row Key 可以提高查询效率，避免热点问题。

数据备份和容错：利用 HDFS 的特性进行数据备份和容错，确保数据的安全性和可用性。

**四、项目实现**

1. 界面实现
2. 主界面



1. 核心功能实现

（1）**网页功能：**

**技术介绍：**

**1.**Vue.js 是一款用于构建用户界面的渐进式 JavaScript 框架。它的核心库专注于视图层，方便与其他库或已有项目进行整合。特点：

响应式数据绑定：Vue.js 提供数据绑定的机制，能自动更新视图。

组件化：支持将界面拆分为可复用的组件，便于维护和管理。

虚拟 DOM：通过虚拟 DOM 提高渲染性能，优化 UI 更新速度。

灵活性：可以与其他库和项目轻松集成，也可以用于构建单页面应用（SPA）。

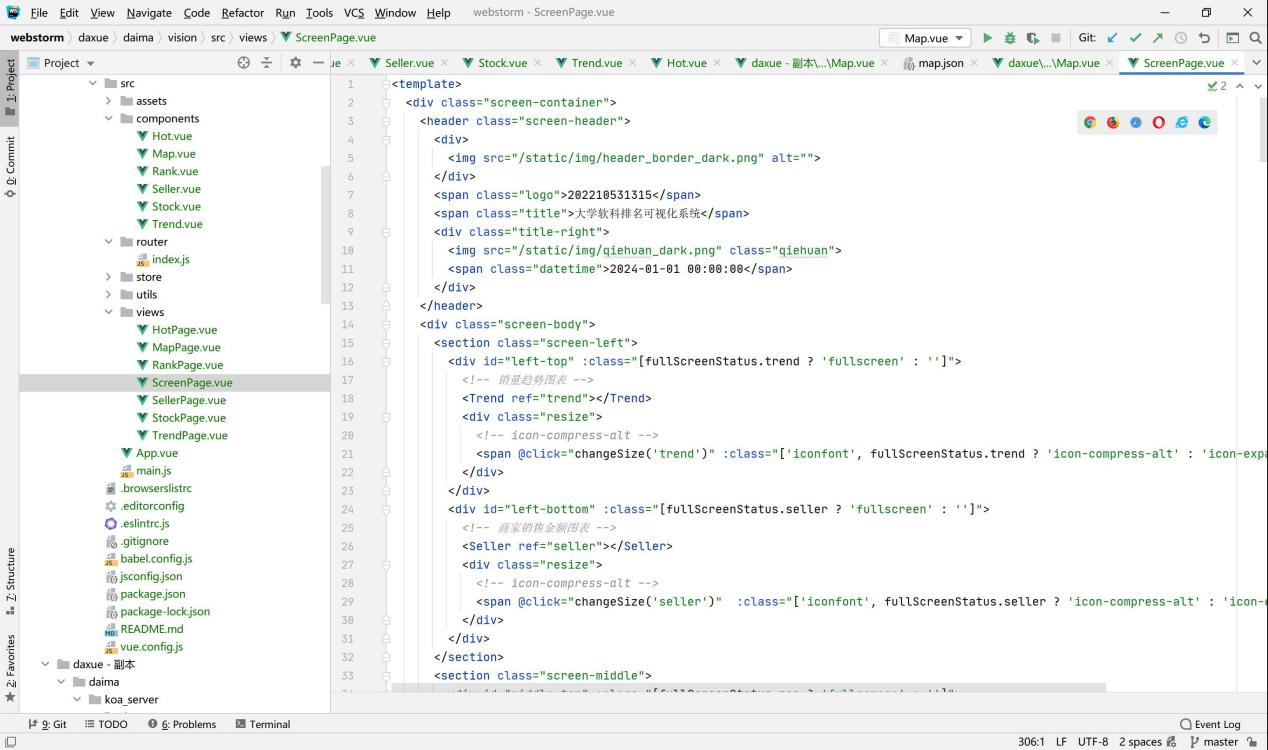
**2.**ECharts 是一个开源的可视化图表库，由百度开发，专注于数据可视化，能够快速生成多种类型的图表。特点：

丰富的图表类型：支持折线图、柱状图、饼图、散点图、雷达图等多种图表类型。

高性能：基于 Canvas 渲染，适合处理大数据量的可视化需求。

交互性：支持丰富的交互操作，如缩放、拖拽、提示框等，增强用户体验。

主题和样式：提供多种主题选择，支持自定义样式。

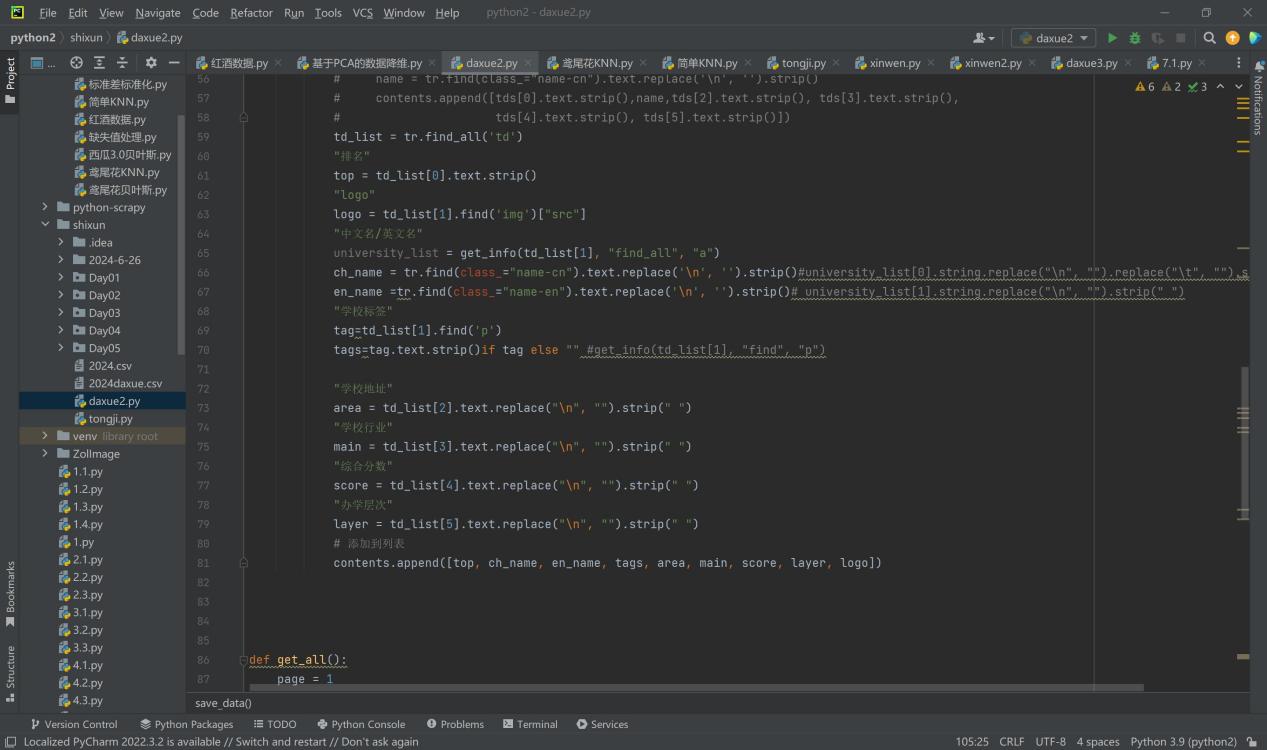


**（2）后端功能：**

**技术介绍：**Python 爬虫是利用 Python 编写的程序，用于自动化地抓取互联网上的数据。它们可以访问网页，提取信息，并将其存储为结构化数据（如 CSV、JSON 或数据库）。

爬虫的基本工作流程通常包括以下几个步骤：发送请求：使用 HTTP 请求（GET 或 POST）访问目标网页。获取响应：接收服务器返回的网页内容（HTML）。

解析数据：使用 HTML 解析库（如 BeautifulSoup 或 lxml）从网页中提取所需信息。存储数据：将提取的数据保存到文件或数据库中。



3.数据库实现

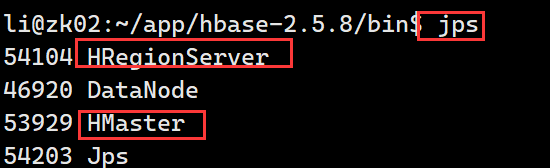
**技术介绍：**在处理大学软科排名数据时，HBase 可以提供以下优势：

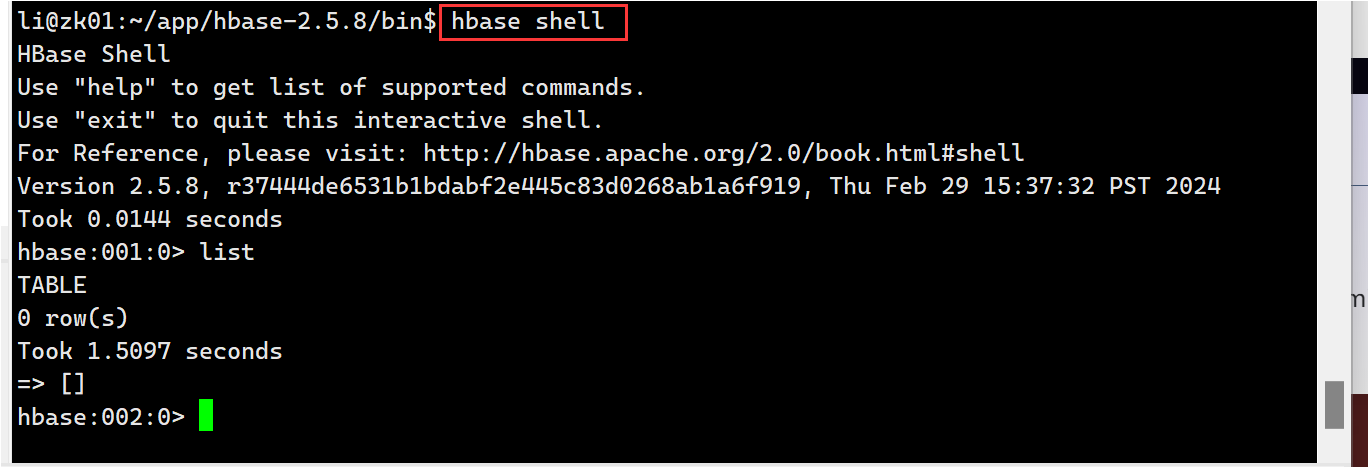
1.大数据处理：大学排名通常涉及大量的指标和数据来源，HBase 能够高效存储和处理这些数据。

2.灵活的数据模型：不同的排名指标可以被存储为不同的列族，使得数据结构灵活且易于扩展。

3.实时分析：对于实时更新的排名数据，HBase 的低延迟读写特性可以支持实时查询和分析。

4.支持大规模并发访问：HBase 可以处理大量的并发请求，适合访问频繁的应用场景。

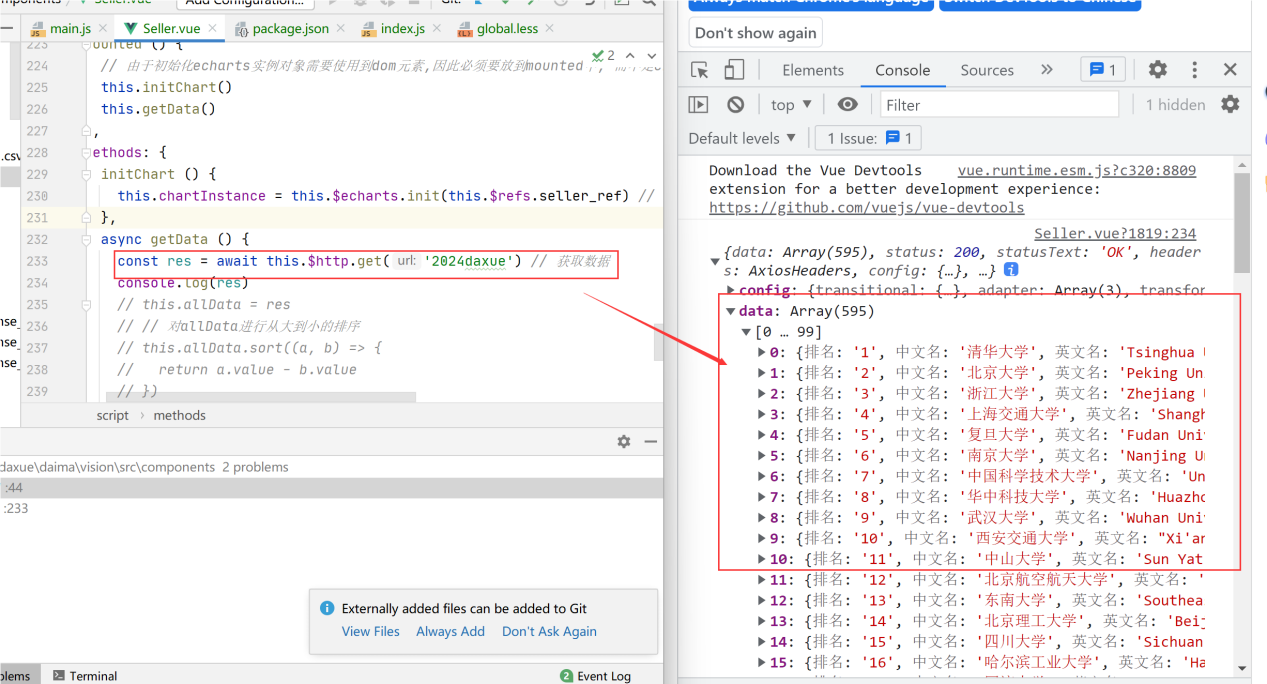




**五、项目测试**

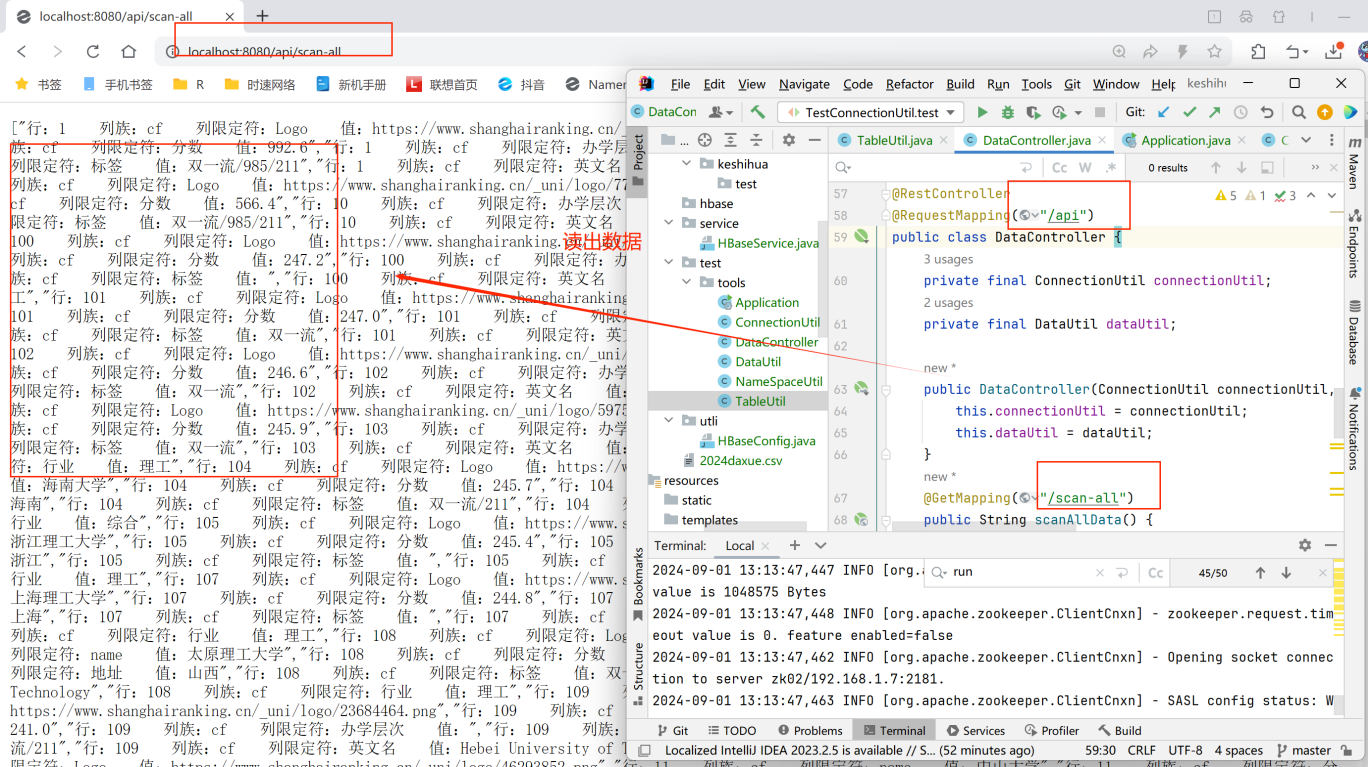
1.单元测试

将hbase数据库映射到本地



2.集成测试

加载数据



3.确认测试

网页界面展示

