

班 级 1903012

学 号 19030100104

西安电子科技大学

# 本科毕业设计论文



题 目 高校招生数据分析软件的设计与实

现

学 院 计算机科学与技术学院

专 业 计算机科学与技术

学 生 姓 名 李晨

导 师 姓 名 谢琨 教授



# 毕业设计（论文）诚信声明书

本人声明：本人所提交的毕业论文《高校招生数据分析软件的设计与实现》是本人在指导教师指导下独立研究、写作的成果，论文中所引用他人的无论以何种方式发布的文字、研究成果，均在论文中加以说明；有关教师、同学和其他人员对本文的写作、修订提出过并为我论文中加以采纳的意见、建议，均已在我的致谢辞中加以说明并深致谢意。

本论文和资料若有不实之处，本人承担一切相关责任。

论文作者：\_\_\_\_\_（签字） 时间：2023年5月25日

指导教师已阅：\_\_\_\_\_（签字） 时间：2023年5月25日



## 摘要

在中国经济社会快速发展、高等教育大众化、新生人口不断下降的背景下，各大高校招生的复杂程度和规模不断提高，对高校招生数据的分析需求也在不断提高。

当前，各大高校都在进行信息化建设。但由于各学校信息化能力参差不齐，以及各高校招生数据的敏感性，目前招生数据分析工作仍以人工分析为主。将招生数据分析工作由传统的人工分析转变为自动分析，这将对高校招生策略的制订、招生质量的提升、招生工作的组织等产生积极深远的影响。

本文基于高校招生工作的实际需求，设计并实现了一款前后端分离的高校招生数据分析软件。该软件使用 Java 语言和 Spring Frameworks 框架进行开发，可以读取高校招生数据，并根据用户需求进行数据分析。在获得分析结果后，可以自动生成数据分析报告。同时，软件前端使用 React JS 框架进行开发，其中页面设计使用了 Semantic UI for React 框架进行辅助设计，在浏览器页面渲染报告供用户预览并提供下载功能。前后端交互使用了 axios 组件发送请求和接收数据。数据可视化方面，使用了 Echarts 组件绘制分析报告相关图表。此外，该软件集成了大规模语言模型 GPT 3.5 Turbo，用于生成个性化的分析报告，并针对该数据进行合理建议，增强了该软件数据分析能力。测试表明，该软件数据分析精准、性能稳定、界面友好，能大幅提升高校招生数据分析工作效能。

关键词： 高校招生，招生数据分析，大规模语言模型



## ABSTRACT

The complexity and scale of enrollment in major universities are increasing under the background of rapid economic and social development, popularization of higher education, and continuous decline in the number of newborns in China. The demand for data analysis of enrollment in universities is also increasing.

Nowadays, major universities are all doing informatization construction. However, analysis work of enrollment data is still mainly done manually because of the uneven informatization capabilities of various schools and the sensitivity of enrollment data. Transforming the analysis of enrollment data from traditional manual analysis to automatic analysis will have a positive and long-term impact on the formulation of enrollment strategies, the improvement of enrollment quality, the organization of enrollment work, etc.

Based on the actual needs of university enrollment work, a front-end and back-end separation enrollment data analysis software was designed and implemented in this article. The software was built by Java programming language and Spring Frameworks. It can read enrollment data of universities and analysis the data according to users' requirements. It can generate data analysis reports automatically after obtaining the analysis results. Meanwhile, the front-end of the software was built by React JS framework. In addition, Semantic UI for React was used to help designing the paper. Reports can be reviewed and downloaded in the browser. Axios component was used to send requests and receive data between the front-end and the back-end. For data visualization, Echarts component was used to render charts in reports. What's more, the software integrated large language model GPT 3.5 turbo to generate personalized analysis reports and provide reasonable suggestions based on the data. It increased the data analysis ability of the software. Tests show that the software can analyse data accurately, have stable performance and friendly interface. It can improve the efficiency of data analysis of university enrollment greatly.

**Keywords:** Higher Education Enrollment, Higher Education Institution Enrollment Data Analysis, Large Language Model





## 目录

摘 要 .....	i
ABSTRACT .....	iii
目录 .....	v
第一章 绪论 .....	1
1.1 课题背景 .....	1
1.2 当前研究现状 .....	2
1.3 本文完成工作 .....	3
1.4 论文结构 .....	3
第二章 数据分析软件相关理论与技术 .....	5
2.1 数据清理技术 .....	5
2.1.1 pandas .....	5
2.1.2 NumPy .....	6
2.2 数据存储技术 .....	7
2.2.1 关系型数据库 .....	7
2.2.2 非关系型数据库 .....	7
2.3 大规模语言模型技术 .....	8
2.4 软件关键技术概述 .....	9
2.4.1 Apache POI .....	9
2.4.2 Spring Boot .....	9
2.4.3 Spring Security .....	10
2.4.4 React JS .....	10
2.5 本章小结 .....	10
第三章 软件需求分析 .....	11
3.1 需求分析 .....	11
3.1.1 功能需求 .....	11
3.1.2 应用需求 .....	13
3.1.3 性能需求 .....	13
3.1.4 安全需求 .....	14
3.2 可行性研究 .....	15
3.2.1 技术可行性 .....	15

3.2.2 经济可行性 .....	16
3.3 本章小结 .....	16
<b>第四章 软件设计与实现</b> .....	<b>17</b>
4.1 软件总体设计与实现 .....	17
4.2 数据预处理工作 .....	18
4.3 数据库设计 .....	20
4.4 数据查询模块设计 .....	22
4.5 数据指标分析计算模块设计 .....	24
4.6 数据可视化模块设计 .....	25
4.7 大规模语言模型辅助模块设计 .....	26
4.8 文档生成模块设计 .....	27
4.9 前端页面设计 .....	28
4.10 本章小结 .....	29
<b>第五章 软件性能分析与测试</b> .....	<b>31</b>
5.1 软件功能测试 .....	31
5.2 软件性能分析 .....	38
5.3 软件性能测试 .....	38
5.4 本章小结 .....	40
<b>第六章 结论</b> .....	<b>41</b>
6.1 软件优点与亮点 .....	41
6.2 软件局限性 .....	41
6.3 软件可优化方向 .....	41
<b>致谢</b> .....	<b>43</b>
<b>参考文献</b> .....	<b>45</b>

## 第一章 绪论

### 1.1 课题背景

随着我国经济和社会的快速发展，人们的生活水平在不断提高，生育观念也在发生着改变。而这一改变，已经体现在了人口数据统计上。即使如今政府已经推出了鼓励二胎政策，一些地方甚至出现了鼓励三胎的政策。但根据国家统计局的数据，2022 年全年人口首次出现下降。中国大陆 2022 年末人口为 14.11 亿，比上年年末减少 85 万人。全年出生人口 956 万人，出生率 6.77‰，自然增长率-0.60‰<sup>[1]</sup>。这是中国社会少子化的征兆。根据多家智库的预测，这种趋势是不可避免的。而少子化带来的，就是未来高考人口的减少。据教育部的估计，2040 年我国参加高考人数预计会减少至 6,000,000 人。而 2022 年我国参加高考人数在 10,300,000 左右，相比 2040 年预估人数降幅达到 40% 左右。伴随着参加高考人数的下降，相对地，尖子生的人数也会相应下降。但于此同时，高校数量，尤其是头部高校数量不会减少。因此，如何在高考人数减少的情况下尽可能多的争夺优秀考生就成为了每个高校都需要面对的问题。

于此同时，高校的信息化进程也任重道远。虽然早在 2001 年 7 月，信息产业部等部门就制定了国家信息化指标体系和企业信息化指标体系<sup>[2]</sup>，教育部也按照信息部的建议要求各大高校加快信息化水平建设。但是，各高校的信息化水平任然参差不齐。一些以文史类见长的高校忽视了信息化的重要性。很多高校的信息化水平还停留在简单的买设备、装网线阶段。信息化水平的参差不齐直接影响了高校的办学效率和行政效率。也直接影响了高校的招生工作。因此，提高高校信息化水平是非常有必要的。

招生工作对一所高校的发展至关重要，而全面精准的招生数据分析可以为高校招生工作的组织和开展奠定良好的基础。而高校招生数据的分析工作甚至是一项更加复杂的工作<sup>[3]</sup>。高校招生数据的分析工作，需要对高校招生数据进行清洗、分析、建模、预测等一系列工作。如果这些工作全部交由人工完成，不仅效率低下，而且可能无法得到正确的结果。由于高校招生数据涉及到考生个人信息，人工分析也会增加安全风险。所以，开发一款高效的高校招生数据分析软件是十分必要的。

1.2 当前研究现状

高校招生数据软件的主要要完成的工作有：数据分析、数据可视化、文档生成等。数据分析是指对数据进行清洗、分析、建模，从而得到数据的特征，并根据数据特征提出合理建议。数据可视化是指将大量复杂的数据和信息以易于交互和易于理解的图表或其他视觉表达形式。文档生成是指根据客制化的包含自定布局的模版和适用于该模版的数据，生成符合客户需求的文档。这些工作在国内外都有一些相关的研究。下面将分别介绍。

国内外已经有很多关于数据分析的理论研究，也基于相关理论开发出了很多数据分析软件。Ralph Kimball 等人提出了数据仓库理念。该理念是组织数据仓库分期和演示领域的集合体，其中运营数据专门用于查询和分析性能以及易用性<sup>[4]</sup>。受到谷歌文件系统和 MapReduce 框架启发开发的 Hadoop 分布式文件系统也可以在分布式系统上高校处理大规模数据。表 1.1 列出了一些常见的数据分析软件。此外亦有学者和开发者基于 Python 研究数据分析方法并开发一些数据分析软件。

表 1.1 数据分析软件排名

软件	受访者	2012	2011
R	245	30.7%	23.3%
Excel	238	29.8%	21.8%
Rapid-IRapid Miner	213	26.7%	27.7%
KNIME	174	21.8%	12.1%
Weka/Pentaho	118	14.8%	11.8%
StatSoft Statistica	112	14.0%	8.5%
SAS	101	12.7%	13.6%
Rapid-IRapid Analytics	83	10.4%	N/A
MATLAB	80	10.0%	7.2%
IBM SPSS Statistics	62	7.8%	7.2%

数据可视化相关技术在国内外亦有充分的研究。大致来说，数据可视化技术的研究方向主要有三个：散点与聚合可视化、线性可视化、分级颜色可视化<sup>[5]</sup>。这些技术可以帮助用户无需学习复杂的数学知识或统计学常识，将高维的复杂数据转化为简单易懂的图表。当前，有许多经过复杂工程考验的稳定的数据可视化工具。比较知名的有：Apache Echarts 等<sup>[6]</sup>。

相比之下，文档生成是一门偏向于工程实践的领域。如今主流的文档格式有：HTML, PDF, doc/docx, ppt/pptx 等。这些格式都有各自的优缺点。一些著名的框架和组件被开发者广泛使用，如：Apache POI, Apache PDFBox, Apache Tika 等。可以根据用户需求自动生成固定格式的文档。此外，也有许多商业化的产品提供 API

服务生成文档并进行格式转换，如 CloudConvert 等。

虽然高校招生数据分析软件所使用到的技术和理论在国内外都有相对充分的研究<sup>[7]</sup>。但是，目前还没有一款专门面向高校的招生数据分析软件。这主要是因为很多高校还没有意识到分析高校招生数据的重要性。找到一款面向考生的报考数据分析软件目前十分容易，在互联网上可以找到很多这样的软件。但是，这些软件与高校招生分析软件的需求和实现方式有很大区别。这也进一步佐证了开发高校招生数据分析软件的必要性。

### 1.3 本文完成工作

本文主要完成了以下工作：

1. 对获得的高校招生数据进行数据清洗，得到了干净的可用于高校招生数据分析系统的数据。
2. 根据高校招生数据特点设计了高校招生数据系统的数据库，确保查询的高效性。
3. 设计了高校招生系统的后端系统，实现了数据分析逻辑和文档生成逻辑，并提供了简洁的 RESTful API。
4. 设计了简洁的前端页面，实现了数据可视化功能和生成文档预览功能。
5. 设计了前后端通信接口，保证了前后端的数据交互的可靠性。
6. 设计了后端系统的安全机制，保证了系统的安全性。

本文还针对实现的高校招生数据分析系统进行了性能测试，撰写了该系统的测试报告和软件文档。

### 1.4 论文结构

本文共分为六章，前两章是对使用技术的概述，第三章至第五章介绍了高校招生分析软件的详细细节，最后一章讨论了该系统的优缺点和后续改进方向。

第一章，简要介绍了开发高校招生数据分析软件的背景和意义，分析了当前研究进展，概括了本文的完成工作。第二章，较为全面的介绍了高校招生数据分析软件所使用到的关键理论和技术。第三章，进行了高校招生数据分析软件的需求分析和科学性研究。第四章，详细介绍了高校招生数据分析系统的设计与实现。第五章进行了软件性能分析与测试。第六章总结该软件的优点和创新点，指出软件当前存在的局限性，并提出未来可优化的方向。



## 第二章 数据分析软件相关理论与技术

### 2.1 数据清理技术

数据清理技术是数据分析中会运用到的一项重要技术，它主要是对数据进行预处理，使得数据能够更好地被分析。数据清理技术主要包括数据去重、数据缺失值处理、数据异常值处理、数据标准化、数据归一化、数据离散化等技术。其中，很多数据分析工作会用到 Python 编程语言。这主要是因为 Python 语言易于上手，而且开源社区发展比较好，拥有很多功能完备、鲁棒性好、文档完备的第三方框架，可以快速进行数据清理工作。接下来会主要介绍数据清洗工作中常用的两个常用框架与技术：pandas 和 NumPy。

#### 2.1.1 pandas

pandas 是 Python 数据分析库的缩写 (Python Data Analysis Library)，是一款为使用 Python 编程语言进行数据操纵和数据分析的软件库。pandas 主要解决了两个 Python 在数据清洗方面的问题：第一，缺乏实现标准模型的库和用于专用模型的内聚框架；第二，缺乏具有集成处理元数据的高级数据结构。<sup>[8]</sup> 这里所说的“元数据”是指为数据点提供标签信息的数据。例如，一些表格会有列的标签信息，对于某些表格，如记录时序数据的表格，这些表格会存在关于行的标签信息，也就是时间戳。这些标签信息可以用于分类和聚合原图表的信息并形成新的数据透视表或者列联表。在 pandas 推出之前，Python 很少有相关库可以处理相关的元数据。

pandas 的主要数据结构是 DataFrame。该数据结构主要借鉴了 R 语言中的 data.frame，内部使用了 NumPy 的数组进行储存数据。每一个轴的 pandas 数据结构都存在一个 Index 对象存储每一轴的标签信息。通常情况下，这些信息是通过 NumPy 的一维数组存储的，表现为一个一维向量。

对于多维数据，pandas 设计了一个新的内部结构 BlockManager 去存储多维数据。因为 pandas 可以支持存储存在多种数据类型的数据在一个 DataFrame，所以 BlockManager 会根据数据类型进行分块存储，每一块使用一个 NumPy 数组进行存储。

为了避免时常使用原生索引方法降低运行效率和程序可读性，pandas 为所有数据结构定义了一个特殊的标签索引属性 ix，减少原生索引的使用。得益于 CPython 等基于 C/C++ 开发的 Python 解释器和扩展，pandas 可以在线性复杂度下完成数据对齐，处理缺失数据等任务。

创造性地，pandas 提出了使用分级索引（Hierarchical Indexing）的数据结构去索引数据<sup>[8]</sup>。这种数据结构可以让一个轴的数据拥有不止一个标签信息。这种数据结构提供了一种在低维数据结构下处理高维数据结构的方法。这大大提高了处理高维数据的效率。此外，pandas 还做了许多针对特殊数据类型的优化工作，使得处理特殊数据类型的效率大幅提高。

截止目前，pandas 已经成为 Python 语言中最受欢迎的数据清洗框架之一。pandas 项目得到了来自 Nvidia, Intel 等大型科技公司的赞助，并且得到了许多来自开源社区的帮助。可以说，pandas 在某种程度上已经成为了 Python 数据分析的事实标准。

2.1.2 NumPy

NumPy 是一个为 Python 编程语言提供对大规模、多维数组和矩阵以及操纵这些数组和矩阵一系列运算的支持的软件库。NumPy 主要针对 Python 编程语言的 CPython 解释器进行了针对性优化。由于 CPython 是一个未优化的字节码解释器，所以基于该解释器编写的数学运算算法对比经过编译的相同算法运行速度慢很多。性能损耗主要是由编译器优化的缺失造成的。<sup>[9]</sup> NumPy 一方面通过提供能在数组上高效运行的多维数组来解决性能损耗问题。另一方面，NumPy 还重写了一系列有关于内部循环的代码实现，提高了算法的运行效率。

NumPy 技术的核心在于提供了一种新的数据结构 ndarray。ndarray 是一个用于处理多维数组的数据结构。不同于 Python 编程语言内置的关于数组的数据结构，ndarray 是数据类型一致的，也就是说 ndarray 中所有元素之间的数据类型都是相同的。这种数据主要使用了表 2.1<sup>[10]</sup> 中的属性描述数组在内存上的分布。通过强制规范数据类型避免了 Python 类型检查时所消耗的性能，提高了运算效率。

表 2.1 ndarray 的属性

Attribute	Description
Data pointer	the memory address of the first byte in the array.
Data type description	the kind of elements contained in the array, for example floating point numbers or integers.
Shape	the shape of the array, for example (10, 10) for a ten-by-ten array, or (5, 5, 5) for a block of data describing a mesh grid of x-, y- and z-coordinates.
Strides	the number of bytes to skip in memory to proceed to the next element. For a (10, 10) array of bytes, for example, the strides may be (10, 1), in other words: proceed one byte to get to the next column and ten bytes to locate the next row.
Flags	which define whether we are allowed to modify the array, whether memory layout is C- or Fortran-contiguous, and so forth.

NumPy 还实现了拓展机制，这个机制可以有效保证数组之间即使纬度不同也可以正确地进行运算。这种机制同样可以减少中间内存的使用，减少运算步骤，从而达到提高运算性能的目的。总的来说，NumPy 运算大多数计算时间都花在矢量化数组操作上，而不是在 Python for-loops 中，而这通常是性能瓶颈。通过优化编译器（如 Cython）实现了进一步的速度改进，这些编译器可以更好地利用缓存



效果。可以说 NumPy 已经成为 Python 数据处理和科学运算的基石和标准。

## 2.2 数据存储技术

数据存储技术是一切数据分析软件和信息管理系统的基石。高效、稳定的数据存储技术决定了数据分析软件和信息管理系统的性能。所以，选择合适的数据存储技术是本文的重要工作之一。通常，数据存储技术存在两大类，一类是关系型数据库，另一类是非关系型数据库。这两种类型各有优势和劣势。这里将分别介绍两种类型的数据存储技术。本项目使用了关系型数据库 MySQL 存储后端数据。这主要是基于数据的规模、内部关系、复杂程度，以及编码的便捷性做出的综合选择。

### 2.2.1 关系型数据库

关系型数据库是一种支持存储和访问互相之间存在关系的数据的数据库。关系模型意味着逻辑数据结构（数据表、视图和索引）与物理存储结构是分开的。这种分离意味着数据库管理员可以管理物理数据存储，而不会影响作为逻辑结构对该数据的访问。例如，重命名数据库文件不会重命名存储在其中的表。关系型数据库在软件开发中已经得到了广泛应用。目前，有很多成熟的关系型数据库，如 Oracle Database, Microsoft SQL server, MySQL, SQLite, MariaDB。这些数据库管理系统针对不同的需求提供了不同的针对性优化。比如，SQLite 就针对移动设备内存容量低，延迟要求高的特点做出了特殊优化。而 MySQL 是一款开源的通用关系型数据库软件。由于它的开源、稳定、高校的特性，本项目采用其作为后端的数据库。

### 2.2.2 非关系型数据库

非关系型数据库提供了与关系型数据库不同的一种存储和访问数据的机制。这种数据库存储数据时并不依赖关系型数据库所依赖的关系型数据的表格关系。通常来说，非关系型数据库有多种类型，如文档型存储、键/值型存储、图数据存储、时序数据存储、对象数据存储、外部索引存储等。目前，已经有许多成熟的非关系型数据库，如 Redis, MongoDB, Neo4j, Elasticsearch 等。

表 2.2 对比了关系型数据库和非关系型数据库的各项特征。

表 2.2 关系型数据库 vs. 非关系型数据库

因素	关系型数据库	非关系型数据库
数据结构	数据由有预置模式的表存储	数据可由多种方式组织
可拓展性	由于要满足 ACID 特性可扩展性有限	高度可扩展性
数据完整性	由强制 ACID 特性保证数据完整	不能保证数据完整性
查询	使用 SQL	不同的 API 或查询语言
灵活性	灵活性有限	高度灵活
性能	对结构化数据和复杂查询高效	对大量非结构化数据和简单查询高效
数据关系	通过表和键形成的关系	无严格数据关系

2.3 大规模语言模型技术

从技术上讲，语言建模技术是使得机器拥有语言智能的主要方法。语言建模技术的发展经历了统计语言模型，神经语言模型，预训练语言模型等阶段<sup>[11]</sup>。其中，预训练语言模型在许多下游任务中表现十分出色。启发了“预训练和微调”的学习范式。经典的适用于大规模语言模型的预训练模型主要是 Transformer<sup>[12]</sup>。图 2.1 展示了 Transformer 的结构<sup>[13]</sup>。大规模语言模型技术就是基于预训练语言模型的理论，通过提高模型的规模和数据的规模，从而提高模型能力。一系列研究已经证实，更大的预训练模型，拥有相似的模型结构，但参数更多，拥有与较小模型截然不同的能力。例如，拥有 175B 参数的 GPT-3，和拥有 540B 参数的 PaLM，拥有更强的处理复杂任务的能力 LLMsurvey。而 2022 年 11 月推出的 ChatGPT，和 2023 年推出的 New Bing, Google Bard 等基于大规模语言模型的商业化产品已经证明了大规模语言模型在解决问题、提供建议方面的巨大潜力。

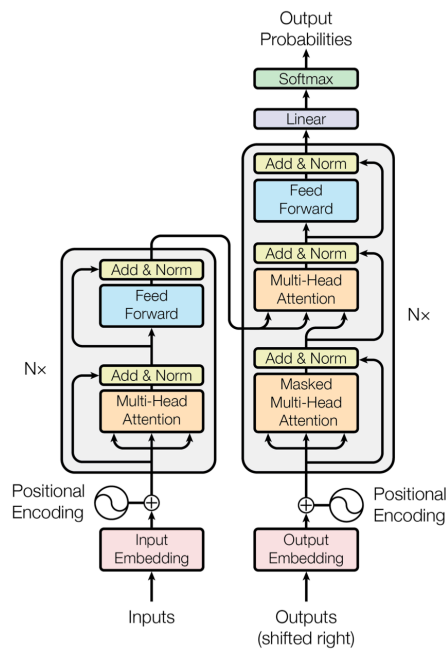


图 2.1 Transformer 模型结构

## 2.4 软件关键技术概述

### 2.4.1 Apache POI

Apache POI 是一个由 Apache Software Foundation 运行的开源项目。而在 Apache Software Foundation 接手前, POI 项目是 Jakarta 项目的一个子项目。Apache POI 为读取和写入 Microsoft Office 格式文件提供了一个纯 Java 的解决方案。现在, Apache POI 已经支持了 .doc/.docx, .xls/.xlsx, .ppt/.pptx 等格式文件的读取和生成。表 2.3 列出了 Apache POI 的主要组件。Apache POI 可以读取和写入 Microsoft Office 格式文件的主要原理是实现了 ISO/IEC 29500:2008 Office Open XML file formats 标准。该标准定义了 Microsoft Office 格式文件的结构和内容。这也就使得即使用户端没有安装 Microsoft Office, 也可以使用 Apache POI 读取和写入 Microsoft Office 文件。

本项目中将主要使用到关于文档和图表生成的 XWPF 和 XSSF 组件。

表 2.3 Apache POI 主要组件

组件名称	描述
HSSF	提供读取和写入.xls 文件的 API
XSSF	提供读取和写入.xlsx 文件的 API
HWPF	提供读取和写入.doc 文件的 API
XWPF	提供读取和写入.docx 文件的 API
HSLF	提供读取和写入.ppt/.pptx 文件的 API

### 2.4.2 Spring Boot

Spring Boot 是 Spring Framework 的一个核心子项目。Spring Framework 是一个为 Java 编程语言编写的反转控制容器和应用框架<sup>[14]</sup>。

Spring Framework 的两个核心特性是面向切面编程框架 (Aspect-oriented programming framework) 和反转控制特性。面向切面编程使得可以将多个类视作一个切面, 从而简化代码逻辑, 提高代码质量和运行效率。反转控制特性使得开发者可以专注于核心代码逻辑的开发, 而将资源和依赖管理交给容器自动完成。

Spring Boot 则在 Spring Framework 上做出了进一步优化<sup>[15]</sup>。其中, 自动配置特性进一步简化了开发者的工作。集成的 Web 服务器也降低了项目部署运行的难度。此外, Spring Boot 还提供了一系列工具提供正在运行的项目信息, 方便开发者调试优化。

### 2.4.3 Spring Security

Spring Security 也是 Spring Framework 中的一个重要组成部分。Spring Security 提供了一系列关于授权和身份认证的 API，使得开发者可以轻松地为项目添加安全控制<sup>[16]</sup>。据一项 2021 年的调查研究，57% 的开发者表示接入 Spring Security 保证项目运行安全是具有绝对高优先级的，且 36% 开发者表示其处在相对高优先级。<sup>[17]</sup>。可以有效防止 CORS 和 CSRF 攻击。

### 2.4.4 React JS

React JS 是由 Meta 推出的开源前端框架。React JS 遵守了指令式编程的范式，简化了前端页面开发的难度<sup>[18]</sup>。同时，也提出了前端组件化的思想，使得前端页面开发工程化。同时提供了多种 Hook 提高前端操纵数据的效率。此外，React JS 运用的 Visual DOM 技术提高了渲染效率，降低了前端性能占用。

## 2.5 本章小结

本章介绍了高校招生数据分析软件开发过程中使用到的关键理论和技术，回顾了这些理论和技术的特点，为后续使用这些理论和技术开发高校招生数据分析软件提供基础。

### 第三章 软件需求分析

软件开发是一整套系统性的工程，其中重要的第一步就是需求分析。广义上的需求分析是指收集软件需求的过程。这个过程包括确定软件应该具有的功能和应该具有的特性。此外，在需求分析的过程中，还需要考虑到这些功能和特性在软件开发过程中的可行性。这些可行性主要包括技术可行性和经济可行性。只有在仔细进行了需求分析之后，才能明白软件的大致架构，从而制定符合需求，且具有可行性的开发方案。使得开发的最终产品满足需求，保持稳定性和扩展性。

#### 3.1 需求分析

##### 3.1.1 功能需求

高校招生数据分析软件主要解决的痛点问题是高校招生数据分析的效率问题。本文在需求分析阶段经过导师的协助，对我校进行招生数据分析工作的流程进行了调研。总结出了传统的高校招生数据分析工作的流程，流程图如图 3.1 所示。

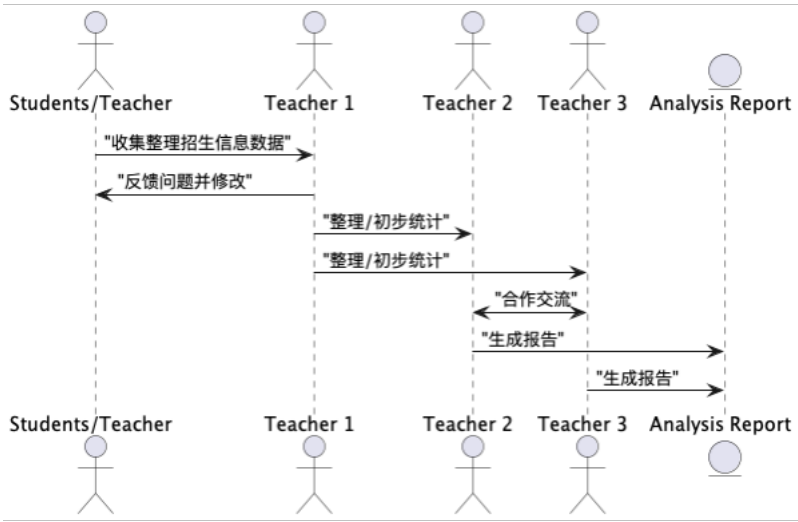


图 3.1 传统高校招生数据分析工作流程

事实上，这个流程图大幅简化了实际过程中的工作量。在实际工作过程中，“合作交流”阶段，“收集整理招生信息数据”阶段和“反馈问题并修改”阶段需要多轮反复进行。这些阶段的工作量要比流程图中展示的更大更复杂。经过分析，发现传统的高校招生数据分析工作流程中存在以下几点问题与不足：

- 1. **人力损耗大** 传统的高校招生数据分析工作流程中，需要多个部门的多个人员参与，这增加了人力资源的浪费；

- 2. **沟通成本高** 传统的高校招生数据分析工作流程中，由于需要多个部门协同进行工作，因此需要部门间多次进行沟通，这增加了沟通成本，降低了工作效率；
- 3. **流程程序长** 传统的高校招生数据分析工作流程中，不同流程由不同的人员参与，且不同流程间人员需要多次沟通，修改，导致流程增长，降低了工作效率；
- 4. **信息化程度低** 传统的高校招生数据分析工作流程中，信息化工具利用率低，更多依赖人工，一些可以交由计算机完成的工作也由人工完成，降低了工作效率。

基于此，笔者对传统的高校招生数据分析工作流程进行了改进，提出了新的基于高校招生数据分析软件的工作流程。新的流程如图 3.2 所示。

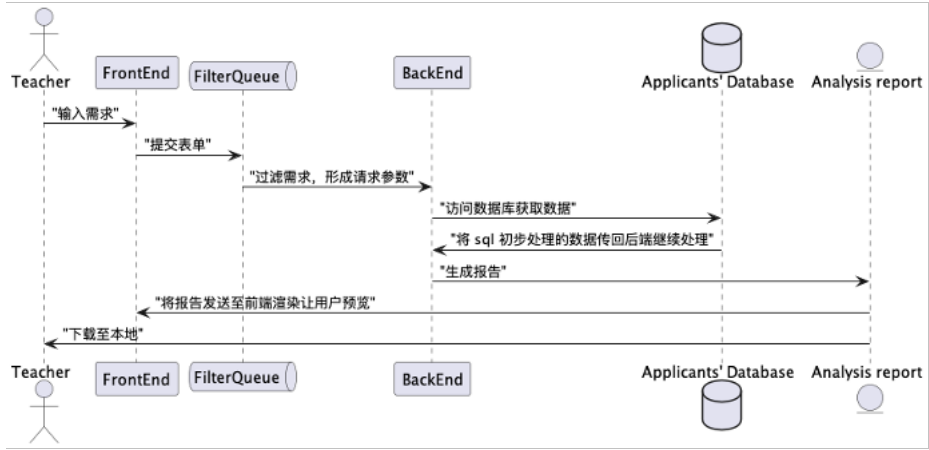


图 3.2 传统高校招生数据分析工作流程

从流程图中，可以看到，在新的工作流程中，高校招生数据分析工作所需要的人员大幅减少了，这减少了人力资源的浪费。同时，新的工作流程中，高校招生数据分析工作所需要配合的部门减少，从而降低了沟通成本。此外，新的工作流程中，将原本需要人工参与的部分交给了软件完成，所短了流程长度，提高了高校招生数据分析工作中的信息化程度和工作效率。

在新的工作流程中，高校招生数据分析软件发挥了重要作用。经过流程的改进和对软件功能需求的进一步分析，高校招生数据分析软件的功能需求如下：

- 1. 数据导入功能：高校招生数据分析软件应该支持从后端数据库中读取数据。同时后端数据库应该方便后端管理员进行数据的导入、更正、删除；
- 2. 数据处理功能：高校招生数据分析软件应该支持对导入的数据进行处理。具体的数据操作包括排序、筛选、分组、聚合、计算等；
- 3. 数据可视化功能：高校招生数据分析软件能够将处理后的数据以图表形式展

示，包括柱状图、折线图、饼图等形式的图表；

4. 数据导出功能：高校招生数据分析软件能够将经过分析的数据导出，方便客户进一步研究使用。

以上是项目初期对高校招生数据分析软件进行功能需求分析得到的功能需求。在开发过程中，这些功能需求指导了软件架构设计、数据库设计、前后端设计等部分。同时，在软件开发过程中，由于一些实际条件的限制和变化，开发过程中最终实现的功能可能会与项目初始阶段功能需求分析得到的功能需求会有微小变化。

### 3.1.2 应用需求

应用需求是指一个软件应该能够满足的应用场景。与功能需求不同，应用需求更关注于软件能够解决的实际应用问题，是对软件更宏观的要求。在高校招生数据分析软件的应用分析过程中，本文对笔者所在学校的高校招生数据分析人员进行了调研。主要调研了高校招生数据分析工作需要完成的目标。经过汇总，高校招生数据分析软件应该满足的应用需求包括：

1. 高校招生数据分析软件应该能够将按照客户需求进行的数据分析结果生成格式化的分析报告，方便客户进一步研究使用；
2. 高校招生数据分析软件应该能够在软件前端页面对生成的数据分析报告进行展示，方便客户在得到分析报告前进行预览，决定继续修改还是确定生成报告；
3. 高校招生数据分析软件应该能够在客户确定软件生成的报告符合要求后生成报告，并提供下载功能，方便客户下载报告至本地进行保存。

高校招生数据分析软件要满足不同层次高校招生数据分析工作的应用需求，设计符合普遍应用需求的前后端系统。同时，在软件设计过程中要预留一定的可扩展性。应对客户可能变化的应用需求。

### 3.1.3 性能需求

分析性能需求在软件开发过程中也十分重要。性能需求是指软件在运行过程中应该满足的性能指标。在性能需求分析过程中，应该考虑到完成软件功能需求和应用需求所需要的最低性能和推荐性能，以及软件目标客户所拥有设备的普遍性能。性能需求分析在软件工程中十分重要。因为只有在满足客户设备性能需求和功能所需最低需求的条件下软件才是可用的。

高校招生数据分析工作一般在高校招生办公室进行。根据多家大学公布的办公室设备配置，以及笔者对笔者所在高校调研结果，总结出了高校办公室设备的

普遍性能，如表 3.1 所示。

表 3.1 高校办公室设备常见配置

组件	型号
CPU	Intel Core I5 (6th gen or newer)
RAM	8GB to 16GB DDR4
存储	256GB to 1TB SSD/HDD
GPU	Intel UHD Graphics / Nvidia GTX 950 or newer
操作系统	Windows 10 professional

接着，我们根据功能需求和应用需求，分析出高校招生数据分析软件的最低配置需求和推荐配置需求。如表 3.2所示。

表 3.2 高校招生数据分析软件推荐配置

组件	型号
CPU	Intel Core I5 or better
RAM	8GB DDR4 or better
存储	至少预留 10 GB 空间
显卡	Nvidia GTX 750 or better
操作系统	Windows 10 or newer

此外，高校招生数据分析软件需要运行设备支持 Java 和 Python 运行环境。满足 MySQL 和 ReactJS 要求的最低配置。通过对比，绝大部分高校办公室设备或高校服务器设备可以部署该高校招生数据分析软件系统。

3.1.4 安全需求

进行安全需求分析对于软件开发十分重要。安全需求分析是指软件应该具有的安全性需求<sup>[19]</sup>。对于数据分析软件等直接接触并操作数据的软件系统来说，安全性对于软件来说更加重要。这是因为客户对于这类软件所操作的数据是敏感的，不希望该软件在接触、操作数据时将数据泄露，或将数据损坏。

对于高校招生数据分析软件系统来说，该软件的安全需求主要体现在以下几个方面：

- 1. 高校招生数据分析软件应该具有鉴权机制，即只有经过授权的用户才能够使用该软件；
- 2. 高校招生数据分析软件应该保证接触的招生数据的完整性，在操作这些数据时不应该直接在数据上进行操作与修改，防止原数据被破坏；



3. 高校招生数据软件需要保证客户与后端通信时的安全性，防止数据在传输过程中遭到窃取或泄露。

对于高校招生数据分析软件来说，安全性需求主要基于高校招生数据的敏感性。高校招生数据是高校内部数据，不应向外界公开。而且，高校招生数据通常包括高校学生的较详细的个人信息。因此，高校招生数据分析软件需要保护高校招生数据的安全。由于要得到较为准确的高校招生数据分析结论需要用到多个高校的招生数据，而这些数据一般存储在各高校的服务器中，并不会暴露在公网环境。所以，需要设计鉴权系统并且对相应数据进行脱敏处理。这样才能保证数据在公网传输过程中的安全性，满足高校招生数据分析软件的安全性需求。

## 3.2 可行性研究

可行性研究在软件开发过程中是非常重要的前期环节。通过可行性研究，可以论证当前软件需求分析中得到的软件应满足的需求是否具有可行性。如果具有可行性，则该需求可以作为软件最终应该实现的功能和特性保留；如果不具有可行性，则应该相应地对需求进行调整，或者放弃该需求。在早期进行详细的可行性研究可以避免在软件开发过程中发现一些需求不可行从而对软件架构等进行大幅修改的情况，提高软件开发效率。可行性研究主要聚焦于技术可行性研究和经济可行性研究。

### 3.2.1 技术可行性

技术可行性研究主要是对软件开发过程中所使用的技术是否具有可行性进行研究。对于高校招生数据分析软件的软件需求来说，大部分需求都具备技术可行性，具有成熟的技术方案可以选择。

对于数据导入功能，可以使用 MySQL 自带的 SQL 编程语言进行数据导入，也可以使用 Python 的 pandas 模块进行数据导入。对于数据预处理功能，可以使用基于 Python 语言的 pandas 和 NumPy 模块进行数据清洗等数据预处理操作。对于数据分析功能，当前各种编程语言都具有相应的 API 可供调用。对于数据可视化功能、文档生成、数据导出功能等也有相应的工具使用。

需要注意的是，用于生成个性化报告的大规模语言模型技术目前拥有两种实现方式：一种是 openai 提供的基于 GPT 3.5 Turbo 的 API；一种是 nomic.ai 提供的基于 GPT-J 模型的方案。前者的模型规模远大于后者（175B vs. 6.5B）。但前者需要消耗大量计算资源，只提供远程访问的 API，加之访问量巨大，会受到网络影响。而后者可以部署在本地，占用资源较少。但后者对中文的支持并不友好。

### 3.2.2 经济可行性

经济可行性在软件开发过程中非常重要。经济上可行对于软件的长久运营和发展起到至关重要的作用。经济上不具备可行性的软件很难取得商业上的成功。

对于高校招生数据分析软件来说,该软件的主要目标客户是各大高校的招生办公室。对于高校招生数据,有一些公开数据可以从互联网获得。而一部分独家数据可以在客户间进行共享。数据的获得并不会成为软件经济可行性的负担。此外,软件使用的技术大部分为开源软件,如 MySQL, Spring Boot 等,这些开源软件的开源协议并不禁止其免费商业使用。所以,技术方面的成本不会成为软件经济可行性的障碍。大规模语言模型的 API 服务需要支付一定的调用费用,这将是经济可行性的一大考量重点。截止目前,GPT 3.5 Turbo 的收费标准为每千 tokens 0.002 美元。这个调用收费标准在软件运营和客户可接受范围之内。

因此,可以说高校招生数据分析软件是具有经济可行性的。其软件运行成本主要体现在数据获取、大规模语言模型 API 调用、服务器运行等方面。这些成本都在可接受范围之内。

## 3.3 本章小结

本章主要叙述了高校招生数据分析软件开发前期的软件需求分析工作。首先,介绍了需求分析的重要性。然后从功能需求、应用需求、性能需求、安全需求四方面对高校招生数据分析软件进行了需求分析得到其应该具备的功能和特性。随后,对高校招生数据分析软件的可行性进行了研究。通过具体分析该技术的路线和各项成本得出该软件具有技术可行性和经济可行性的结论。

## 第四章 软件设计与实现

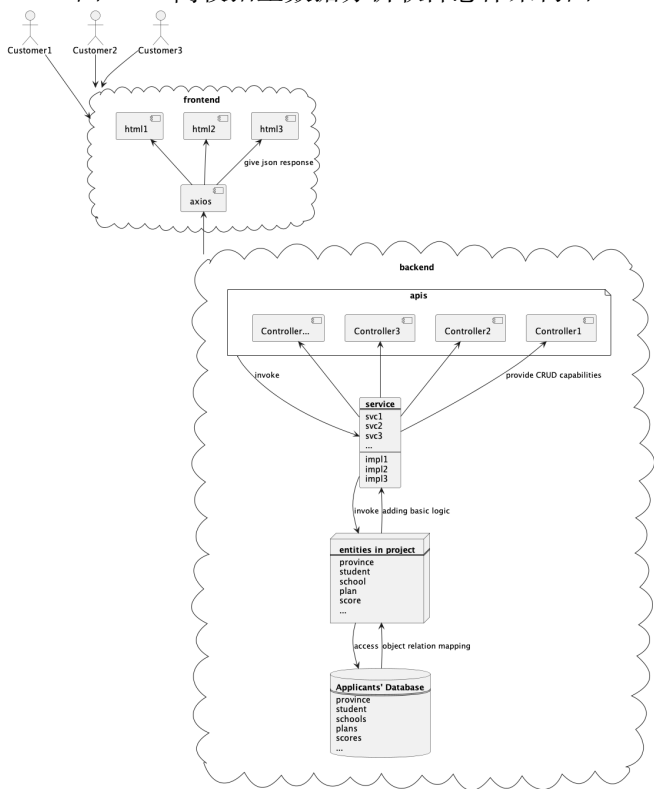
本章将重点介绍高校招生数据分析软件的具体设计与实现，按照软件开发流程和从后端到前端的介绍顺序具体介绍高校招生数据分析软件。

### 4.1 软件总体设计与实现

经过笔者的调研和对目标客户及高校招生数据分析软件所做的软件需求分析，可知高校招生数据分析软件与数据交互的功能主要集中在数据查询模块。相应的数据预处理工作，数据导入和删除工作通常在部署软件时进行，在软件运行时通常不会进行数据的导入和删除。于此同时，大多数目标客户希望软件能够部署在高校本地的服务器或高校负责相关工作的招生办公室的工作用设备上。因此，高校招生数据分析软件非常适合采用 B/S 架构进行开发。B/S 架构，即浏览器/服务器架构，是一种网络架构模式，它将软件系统分为客户端和服务端。这种架构模式非常适用于可小型化部署软件和单体运用的场景。同时，B/S 架构的软件系统可以通过浏览器进行访问，不需要安装客户端，因此适用于移动办公。

所以，高校招生数据分析软件的总体架构设计如图 4.1 所示。

图 4.1 高校招生数据分析软件总体架构图



高校招生数据分析软件使用了经典的 B/S 架构。对于服务器端，主要可以分成四个大的模块：

1. **数据库** 数据库层保存有高校招生数据分析软件所依赖的高校招生数据以及在软件运行过程中生成的一些中间信息和日志；
2. **数据实体访问类** 数据实体访问类是介于应用和数据库之间的抽象层。数据可视化功能访问对象是一种开发模式，通过这种开发模式可以减少 SQL 代码的编写，降低软件开发复杂度。数据实体访问类需要对象关系映射框架或持久层框架的支持；
3. **服务层** 服务层是基于数据实体访问类所实现的关于该数据实体访问类的一些功能，通常包括对该数据实体访问对象的 CRUD 操作；
4. **API 层** API 层通过使用服务层提供的服务，实现该软件系统服务端应该实现的复杂业务逻辑，同时将这些复杂的功能通过 API 暴露给调用者。对于用户来说，只有 API 层是运行时对用户可见的；

相比服务器端，浏览器端，也就是所谓的“前端”的设计相对简单。前端架构主要包括以下三个部分：

1. **前后端交互模块** 前后端交互模块可以通过 axios http 库根据用户需求向服务器端发送 http/https 请求，并将返回内容通过 json 或字节串的方式进行表示，同时也支持并发式请求任务，提高前端效率；
2. **前端路由模块** 前端路由模块负责不同页面之间的路由逻辑，方便不同页面之间进行跳转；
3. **前端组件模块** React JS 支持前端组件式开发，通过开发可复用的组件可以降低开发工作量，同时简化前端页面的数据模型逻辑；

除此之外，在服务器端和浏览器端，还需要一些配置文件和工具类文件，辅助实现业务逻辑。比如，在服务器端，需要生成文档、图表、和使用大规模语言模型技术生成个性化文本的工具类，以及 Spring Security 等需要的配置类。

## 4.2 数据预处理工作

数据预处理工作主要是指对获得的高校招生数据进行清洗和整理。高校招生数据主要分为两部分。主要部分为可从公开渠道查询的高校招生总体数据；另一部分为各高校具体的招生数据，这部分数据因为涉及到考生个人信息等敏感信息不会对外开放。前一部分数据数量相对庞大复杂，而后一部分数据数量相对较少。

数据预处理工作基于 pandas 和 NumPy 对高校招生数据中的大量复杂的公开数据进行清洗整理。这部分公开数据来自于项目合作企业的数据仓库。该数据仓

库为.sql 格式文件，文件大小 1.58GB。该数据仓库原本为为一所学校智慧学校建设的信息系统的数据仓库。除部分公开高校招生数据以外，还包括大量该校智慧办公的其他数据。需要对其进行数据预处理工作。

该数据仓库名称为“zsdatak\_v1”，包含 136 个表。其中，包含多个与高校招生数据无关的表。这些表主要为支撑高校智慧办公系统其他办公逻辑的辅助表。这些表中，名称以“act”，“data”，“gen”，“iim”，“oa”，“qrtz”，“sys”，“test”的多个表为与高校招生数据无关的表。例如，以 test 前缀命名的表用于系统测试，以 qrtz 前缀命名的表用于系统定时任务。这些表占据总表数量的绝大多数。经过分析，存储高校招生数据的表主要有四个，如表 4.1 所示。

表 4.1 原始数据库与高校招生数据相关表

表名	描述
table_yssj_gxxx	高校信息库
table_yssj_sftdfs	省份投档分数
table_yssj_zsjh	招生计划
table_yssj_zyfs	专业分数

在确定了高校招生数据相关表后，可以将相应表单独导出为一个专用于高校招生数据分析的数据库。但是，对这些表进行进一步数据预处理工作是必要的。这些表存在以下三个问题：

- 1. **表名命名不规范** 表名使用了中英文混杂的命名方式，不利于理解和后续软件开发；
- 2. **列名命名不规范** 列名使用了中文拼音缩略的命名方式，同时无说明文档，无法做到自解释；
- 3. **存在冗余字段** 表中存在一些辅助字段，这些字段在高校招生数据分析中没有必要；

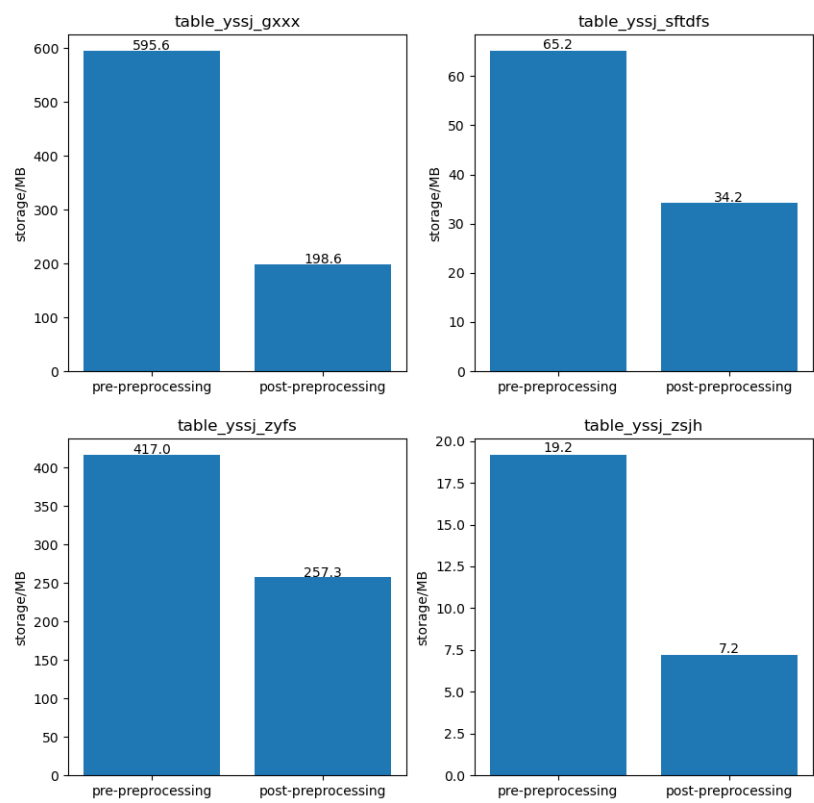
针对表中存在的问题，确定了对表进一步进行数据预处理的流程：

- 1. **表名命名规范化** 由于 MySQL 仅支持英文小写字符命名表名，因此对表的重新命名使用英文命名方式，这样做不仅避免可能出现的兼容性问题，而且可以做到表名自解释，方便软件后续开发；
- 2. **列名命名规范化** 与表名命名规范化相似，将列名使用英文命名方式，做到列名自解释；
- 3. **剔除冗余字段** 将各个表中与高校招生数据分析流程无关的字段，减小高校招生数据库存储占用；

根据确定的数据预处理流程，对高校招生数据库进行了数据预处理，并对数

据预处理前后的内存占用进行了统计。结果如图 4.2 所示。

图 4.2 数据预处理前后内存占用对比



综合来看，经过数据预处理后的高校招生数据库占用内存约 795MB。相比原始数据库的内存占用（1.58GB），减小约 51%。数据预处理工作大幅减小了内存占用，提高了高校招生数据分析软件的运行效率。

### 4.3 数据库设计

高校招生数据主要存储于 MySQL 数据库中。选择 MySQL 主要有两个原因：第一，从高校招生数据方面看，高校招生数据具有较强的关系特征，适合使用关系型数据库存储；第二，MySQL 开源且查询效率高，可以减小高校招生数据分析软件在数据查询期间的性能损耗，提高高校招生数据分析软件的运行效率。

高校招生数据库由六个表组成，如表 2.2 所示。

不同表间通过外键进行关联。其中，province 表中的 province\_id 字段，school 表中的 school\_id 字段，以及 majorscore 表中的 id 字段，负责作为其他表的外键，负责约束表间关系。

表 4.2 高校招生数据库表信息

表名	描述
student	高校入学学生信息
school	采集高校信息
province	省份
majorscore	专业分数统计信息
scorestatistics	分数统计信息
enrollmentstatistics	参加考试人数信息

其中，各表各自拥有的字段如图 4.3 所示。

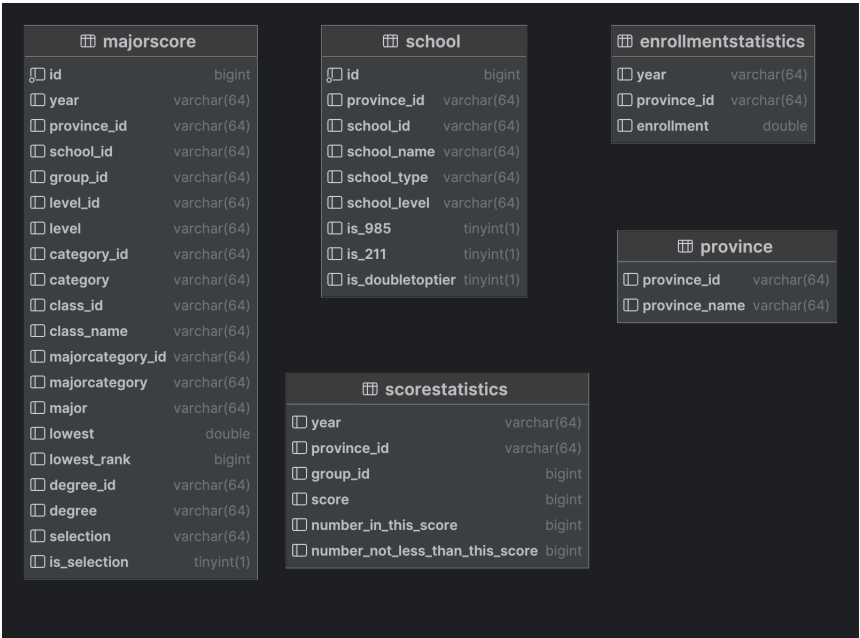


图 4.3 高校招生数据库表字段信息

其中大多数字段名都是可自解释的。例如 year, id, province\_id, school\_id 等字段。另外一些字段名则需要进行相应解释，如表 4.3 所示。

表 4.3 高校招生数据库表部分字段解释

字段名	描述
school_type	高校类型（公办/民办/中外合作办学）
school_level	高校级别（本科/专科）
is_doubletoptier	是否为“双一流”高校
category	专业类别（普通类/定向类/...）
class_name	专业门类（教育部规定的十四种学科门类）
majorcategory	专业大类（由学校划分的专业大类）
selection	选考科目信息（针对实行新高考地区）

原数据中在专业分数统计表中有最高分、平均分、最低分三个统计指标，但由于原数据中大部分记录中最高分和平均分数据缺失，所以在高校招生数据分析软件数据库中没有包含最高分和平均分两项数据指标。

经过数据预处理和关系分析重新设计的高校招生数据库。表间关系清晰，表名和字段名可做到自解释，在需要进一步解释的字段名也提供了相应的说明文档。同时，清除了冗余信息，减小了数据库内存占用。

#### 4.4 数据查询模块设计

数据查询模块模块是高校招生数据分析软件的核心模块。数据查询模块负责按照设定的条件查询高校招生数据库，并返回相应的结果。针对高校信息数据库中的六个表，数据查询模块设计了六个主要的数据查询模块。

针对省份表 province，设计了如下三个数据查询 API：

1. listAll(): 返回数据库中所有省份信息；
2. getProvinceById(id): 根据省份 id 返回相应的省份信息；
3. getProvinceByName(name): 根据省份名称返回相应的省份信息。

省份表是高校招生数据库中的重要辅助表，主要用于辅助高校信息表和专业分数统计表的查询。在许多业务逻辑的实现中都需要使用到关于省份表的数据查询 API。

针对高校信息表 school，设计了如下八个数据查询 API：

1. listAll(): 返回数据库中所有高校信息；
2. list985(): 返回所有“985”高校信息；
3. list211(): 返回所有“211”高校信息；
4. listDoubleTopTier(): 返回所有“双一流”高校信息；
5. listByProvince(prov\_id): 根据省份 id 返回相应的高校信息；
6. getSchoolByName(school\_name): 根据高校名称返回相应的高校信息；
7. getSchoolById(school\_id): 根据高校 id 返回相应的高校信息；
8. searchSchoolsByStr(str): 根据关键词 str 返回相应的高校信息。

这里重点介绍最后一个 API: searchSchoolsByStr(str)。这个 API 可以通过高校名称中的部分搜索到所有与该部分匹配的高校信息。例如，当用户输入“电子科技大学”时，返回结果会包括“电子科技大学”、“西安电子科技大学”等信息。该 API 内部基于 KMP 算法实现了高效的针对高校名称的匹配算法。同时，高校信息库内部通过 B 树结构组织数据，提高了查询效率。此外，当用户输入部分常见于高校名称的字符串，如省份名称“山西”、“河北”，或者表示学校性质的“大学”、“学



院”等时，将返回多个高校信息。而响应信息过多会降低响应速度。为了提高响应速度，该 API 算法设置当满足条件的结果多于 8 个时，会返回前 8 个结果，而不是全部作为响应返回。这样做提升了用户的使用体验。

针对专业分数统计信息表 `majorscore`，设计了如下数据查询 API：

1. `listAll()`: 返回数据库中所有专业分数统计信息；
2. `listMajorBySchool(school_id)`: 根据高校 id 返回相应的专业分数统计信息；
3. `listMajorByProvince(prov_id)`: 根据省份 id 返回相应的专业分数统计信息；
4. `listMajorByYear(year)`: 根据年份返回相应的专业分数统计信息；
5. `listMajorBySchoolIdProvinceIdYear(school_id, prov_id, year)`: 根据高校 id、省份 id 和年份返回相应的专业分数统计信息；

高校专业分数统计信息表是高校招生数据的主体部分，保存了高校在各个省份的各个专业的招生信息。在许多业务逻辑的实现中都需要使用到关于专业分数统计信息表的数据查询 API。

针对参加考试人数信息表 `enrollmentstatistics`，设计了如下数据查询 API：

1. `listAll()`: 返回数据库中所有参加考试人数信息；
2. `listEnrollmentByProvince(prov_id)`: 根据省份 id 返回相应的参加考试人数信息；
3. `listEnrollmentByYear(year)`: 根据年份返回相应的参加考试人数信息；

参加考试人数信息表是高校招生数据分析的重要辅助表。一些业务逻辑需要使用到参加考试人数信息。

针对分数统计信息表 `scorestatistics`，设计了如下数据查询 API：

1. `listAll()`: 返回数据库中所有分数统计信息；
2. `listScienceByYearAndProvince(year, prov_id)`: 根据年份和省份 id 返回相应的理科分数统计信息；
3. `listArtByYearAndProvince(year, prov_id)`: 根据年份和省份 id 返回相应的文科分数统计信息；

分数统计信息表是高校招生数据分析的重要辅助表。一些业务逻辑需要使用到分数统计信息。需要注意的是，一些省份在最近年份改革了高考制度，取消了传统的文理分科，因此在这些省份中，API 对这些省份的分数统计信息进行了针对性的判定和处理。

针对学生表 `student`，设计了如下数据查询 API：

1. `listAll()`: 返回数据库中所有学生信息；
2. `listStudentBySchool(school_id)`: 根据高校 id 返回相应的学生信息；

3. listStudentBySchoolAndMajor(school\_id, major\_id): 根据高校 id 和专业 id 返回相应的学生信息;
4. listStudentBySchoolAndMajorAndGender(school\_id, major\_id, gender): 根据高校 id、专业 id 和性别返回相应的学生信息;
5. listStudentByName(name): 根据学生姓名返回相应的学生信息;

需要注意的是, 学生姓名可能会存在重复, 所以通过姓名查询学生信息时返回的学生信息以数组形式返回。

通过上述设计的 API, 高校招生数据分析系统可以方便的调用数据库中的数据, 方便进行进一步的数据分析。

#### 4.5 数据指标分析计算模块设计

高校招生数据分析不仅需要调用数据, 而且需要将调用数据进一步计算相应指标。根据相关指标, 得到与高校招生相关的结论。

经过总结相关文献, 调研高校招生工作分析流程, 高校招生数据分析系统设计了如下指标:

1. 按志愿录取率;
2. 生源指数;
3. 全国平均排名;
4. “千量” 指数。

按志愿录取率是针对专业受欢迎程度的统计指标。设该专业录取总人数为  $p$ , 该专业录取学生中设置被录取专业为志愿专业人数为  $q$ , 则该专业的按志愿录取率可由如下公式计算:

$$\text{按志愿录取率} = \frac{q}{p} \quad (4-1)$$

按志愿录取率越高, 说明该专业受欢迎程度较高, 更受考生认可。分析历年各专业的按志愿录取率可以反映该专业招生工作的发展趋势。例如, 对某大学的网络空间安全专业统计 2018 年-2020 年的按志愿录取率。2018 年按志愿录取率为 61.67%, 2019 年为 86.67%, 2020 年为 100%。可以看出该专业的受认可程度逐渐上升。

生源指数是针对专业生源质量的统计指标。设各专业所在省份招生最低分在该省的排名为  $p_i$ , 则生源指数可由如下公式计算:

$$\text{生源指数} = \frac{\sum_{i=1}^n p_i}{n} \quad (4-2)$$

生源指数越低, 说明该专业生源质量越高。分析历年各专业的生源指数可以反映

该专业生源质量的发展趋势。

全国平均排名是针对该高校在全国的排名的统计指标。设各专业所在省份招生最低分在全国的排名为  $p_i$ ，则全国平均排名可由如下公式计算：

$$\text{全国平均排名} = \frac{\sum_{i=1}^n p_i}{n} \quad (4-3)$$

全国平均排名越低，说明该专业在全国的排名越高。分析历年各高校的全国平均排名可以反映该高校在全国的排名的发展趋势。

“千量”指数是针对该高校招生质量的统计指标。设该高校各专业录取最低分的最高值为  $s_{max}$ ，最低值为  $s_{min}$ ，则“千量”指数可由如下公式计算：

$$\text{“千量”指数} = \frac{s_{max} - s_{min}}{1000} \quad (4-4)$$

“千量”指数越低，说明该校招生的平均生源水平约接近，招生质量越高。

高校招生数据分析系统针对提出的指标设计了相应的 API，方便客户快速进行数据分析。API 的设计如下：

1. enrollmentRateByMajor(school\_id, major\_id, year): 返回某高校某专业某年的按志愿录取率；
2. studentQualityIndex(school\_id, major\_id, year): 返回某高校某专业某年的生源指数；
3. averageRanking(school\_id, year): 返回某高校某专业某年的全国平均排名；
4. thousandIndex(school\_id, prov\_id, year): 返回某高校某年在某省份招生的“千量”指数；

## 4.6 数据可视化模块设计

在数据可视化模块，高校招生数据分析软件设计了三种可视化图表：柱状图、折线图和饼图。在设计数据可视化模块时，高校招生数据分析软件使用了 Java-Echarts 库进行绘图。Java-Echarts 是针对 Echarts 的 Java 封装，可以使用 Java 快速生成 Echarts option 对象。

高校招生数据分析软件使用 Java-Echarts 根据数据分析信息生成 Echarts option 对象，然后将该对象转换为 JSON 字符串，返回给前端页面。前端页面的 Echarts 组件可以 Echarts option 对象为输入，绘制出相应的图表。设计 API 如下：

1. drawIntBar(String title, String[] xAxis, Map<String, Integer[]> series, String html-Path): 根据信息绘制柱状图，数据类型为整数；
2. drawFloatBar(String title, String[] xAxis, Map<String, Double[]> series, String htmlPath): 根据信息绘制柱状图，数据类型为浮点数；

3. `drawIntLine(String title, String[] xAxis, Map<String, Integer[]> series, String htmlPath)`: 根据信息绘制折线图, 数据类型为整数;
4. `drawFloatLine(String title, String[] xAxis, Map<String, Double[]> series, String htmlPath)`: 根据信息绘制折线图, 数据类型为浮点数;
5. `drawIntPie(String title, Map<String, Integer> items, String htmlPath)`: 根据信息绘制饼图, 数据类型为整数;
6. `drawDoublePie(String title, Map<String, Double> items, String htmlPath)`: 根据信息绘制饼图, 数据类型为浮点数;

这些 API 不仅可以帮助前端进行可视化, 同时可以在后端本地生成图片, 辅助后端进行文档生成。

#### 4.7 大规模语言模型辅助模块设计

高校招生数据分析软件使用大规模语言模型技术辅助文档生成。为了保证稳定性和辅助生成文档的质量, 选用 GPT 3.5 Turbo 模型进行文档生成是相对合适的选择。

大规模语言辅助模块的 API 设计相对简单, 只有一个 API: `getAnswer(String question)`。将按照格式生成的文件作为提示词上传, 即可得到个性化的文档段落。

大规模语言模型辅助模块的工作流程如图 4.4 所示。

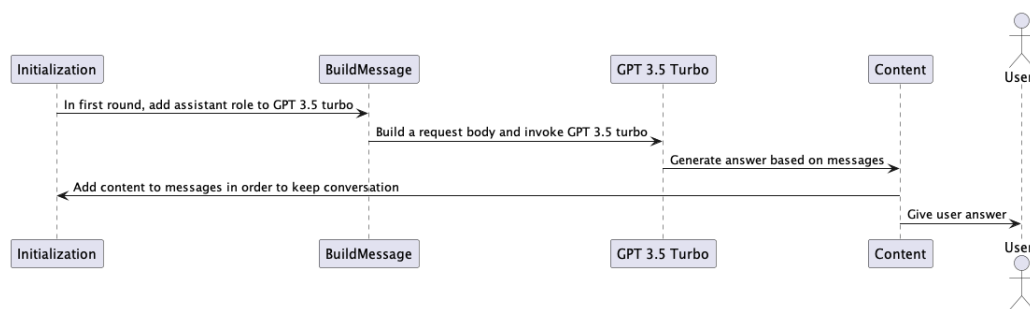


图 4.4 大规模语言模型辅助模块工作流程

大规模语言模型辅助模块工作效率受 prompt 的影响很大。如果 prompt 的指令不够明确, 则生成的分析文本的质量不会稳定。

为了保证分析文本的生成质量, 我根据 prompt engineering 的相关原则 (指令、情景、数据), 提出了构建 prompt 的方法:

1. **指标概述** 将指标的名字和含义概括性的介绍, 并且给出指标的意义;
2. **指标具体描述** 给出指标趋势代表的含义, 以及趋势变化能够反映的现象;
3. **指标定义式** 给出指标的定义计算公式;

- 4. **具体指标** 给出需要分析的具体指标；
  - 5. **分析要求** 给出根据上述描述和数据进行分析的具体要求；
- 例如，对于以下专业全国平均排名数据，可以构建出结构化 prompt。

表 4.4 专业全国平均排名数据示例

年份	全国平均排名
2017	8575.43105
2018	6666.74175714
2019	6375.215275
2020	4595.42857333
2021	4841.93817097

- 1. ”全国平均排名”指数是根据历史经验总结的一个适用于分析学校招收热门专业的指标。对于分析招生工作效果十分有参考意义。
- 2. ”全国平均排名”指数的趋势可以看出某学校专业的受欢迎程度。通常来说，该指标值越低越好（该值越低说明排名越靠前）。如果该值呈现下降趋势说明该专业正在越来越受欢迎。
- 3. “全国平均排名指数 “的定义是” 全国平均排名指数 = 各省专业最低排位的平均值 “。
- 4. 西安电子科技大学计算机专业的“全国平均排名”分别是: 2017 年 8575.43105 2018 年 6666.74175714 2019 年 6375.215275 2020 年 4595.42857333 2021 年 4841.93817097。
- 5. 请你用生动的语言对这些数据进行分析并得出一些趋势和结论。

通过实验证明，对于同一份文档，基于 prompt engineering 提出的具有“指令、情景、数据”的结构化 prompt 可以稳定大语言模型输出质量，同时得到符合要求的更高质量文本。

4.8 文档生成模块设计

文档生成模块是高校招生数据分析软件的重要工具模块。负责将高校招生数据的分析结论生成格式化的文档报告。

文档生成模块的 API 设计如下：

- 1. addTitle(String title): 为文档添加标题；
- 2. addHeading(String heading): 为文档添加一级大标题；
- 3. addSubHeading(String heading): 为文档添加二级标题；
- 4. addParagraph(String text): 为文档添加段落；

5. addQuote(String quote): 为文档添加引用;
6. addImage(String imgPath) throws Exception: 为文档添加图片;
7. addTable(int column, String[] columnNames) throws Exception: 为文档添加表格;
8. addTableItem(String[] item) throws Exception: 为文档添加表格项;
9. save() throws FileNotFoundException, IOException: 保存文档至本地路径。

需要注意的是,若文档可以转换为 PDF 格式,则需在创建文档时指定文档风格,确定文档页面尺寸。设置文档风格的代码如下:

```
1 XWPFStyles styles = this.document.getStyles();
2
3 // there must be section properties for the page having at least
  the page size set
4 CT SectPr sectPr = this.document.getDocument().getBody().
  addNewSectPr();
5 CTPageSz pageSz = sectPr.addNewPgSz();
6 pageSz.setW(BigInteger.valueOf(12240)); //12240 Twips = 12240/20 =
  612 pt = 612/72 = 8.5"
7 pageSz.setH(BigInteger.valueOf(15840)); //15840 Twips = 15840/20 =
  792 pt = 792/72 = 11"
```

上述代码将文档页面尺寸设置为 8.5 英寸 × 11 英寸,约为 A4 页面大小。

## 4.9 前端页面设计

前端页面使用标准的 React JS 项目框架。由于高校招生数据软件的业务场景相对精准单一,所以前端页面设计相对简单。

前端页面主要包括三大组件:

1. 登录组件: 负责在浏览器未保存登录鉴权信息时,向后端发送登录请求,获取登录鉴权信息;
2. 主页组件: 负责在拥有鉴权信息后,在主页输入高校招生分析需求,并将需求发送回后端进行数据分析;
3. 预览组件: 负责在数据分析完成后,将分析报告在前端展示给用户预览,以便做后续决定。

前端界面使用 Semantic UI for React 进行设计,大大减少了前端 CSS 格式设计工作。同时,其支持响应式页面,因此在多种设备的尺寸下都有良好的视觉表现。前后端交互使用了 axios http request library。axios http request 支持并发式处理 http request,方便多个终端同时访问服务<sup>[20]</sup>。

针对预览组件,在服务器端设计了.docx 转换到.pdf 格式的 API。由于 Apache

POI 早期开发时项目流程不规范，以及后期多次 API 大幅变动，将 Apache POI 文档转换为 PDF 需要对转换器 converter 进行配置才能正常渲染简体中文。对 converter 进行配置的代码如下：

```
1 Options options = Options.getFrom(DocumentKind.DOCX).to(
  ConverterTypeTo.PDF);
2
3 PdfOptions pdfOptions = PdfOptions.create();
4
5 pdfOptions.fontProvider(new IFontProvider() {
6     @Override
7     public Font getFont(String familyName, String encoding, float
  size, int style, Color color) {
8         try {
9             BaseFont bfChinese = BaseFont.createFont("/System/
  Library/Fonts/Supplemental/Arial Unicode.ttf",
10                BaseFont.IDENTITY_H, BaseFont.EMBEDDED);
11             Font fontChinese = new Font(bfChinese, size, style,
  color);
12
13             if (familyName != null) {
14                 fontChinese.setFamily(familyName);
15             }
16
17             return fontChinese;
18         } catch (Exception e) {
19             System.out.println(e.getLocalizedMessage());
20             return ITextFontRegistry.getRegistry().getFont(
  familyName, encoding, size, style, color);
21         }
22     }
23 });
24
25 options.subOptions(pdfOptions);
```

其中第 9 行代码中，字体路径需要根据配置的服务器或其他设备环境进行修改。

## 4.10 本章小结

本章主要介绍了高校招生数据分析软件的系统设计。首先介绍了系统的总体架构。接着介绍了对高校招生数据的数据预处理工作，这对高校招生数据软件开发非常重要。然后介绍了系统的后端设计，包括数据查询模块、数据分析模块、文档生成模块、大规模语言模型辅助模块。最后介绍了系统的前端设计，包括前端页面设计、.docx 文件转换为.pdf 文件的 API 设计。





## 第五章 软件性能分析与测试

软件开发过程中和软件开发完成后都需要定时对软件进行分析与测试，及时发现软件存在的漏洞和存在的性能瓶颈，以便及时修复和优化。通常，在软件开发过程中和软件完成并交付后需要进行软件性能分析和测试。软件性能分析是在软件部署前对软件进行性能分析，预测软件各部分的性能，并分析软件可能存在的软件性能瓶颈。软件性能测试是设计合理的软件测试流程，对软件各项功能进行测试，发现存在的问题。并得出相应的测试数据。

### 5.1 软件功能测试

针对本文中设计和实现的高校招生数据分析软件，笔者设计了通过上线实际测试的流程测试各项功能是否可正常运行。

高校招生数据分析软件由于接触到一些高校不愿公开或者涉及个人信息的数据，因此保证数据安全十分重要。因此，高校数据分析软件设计了账户系统。账户需要由高校管理员在部署高校招生数据分析软件时进行创建。具体创建配置在 application.yml 文件中。遵循 Spring Security 的账户创建协议。

当项目部署完成后，高校招生数据分析人员进入系统后，会进入高校招生数据分析软件登录页面，该页面的路径为”<deploy ip address>:3000/login”。登录页面如图 5.1 所示。



图 5.1 登录页面

执行登录操作时，前端会向服务器发送经过 AES 加密的含有账户密码的

POST 请求。当登陆成功时，服务器端会发送给前端一定时间（一个 HTTP Session）的 token 并让前端浏览器保存在本地。前端会经过 React Router 进入高校招生数据分析软件主页；当登陆失败时，主页会出现错误提示，如图 5.2 所示。同时会重定向至登录页面进行登录。



图 5.2 登录错误提示

主页操作界面如图 5.3 所示。主页包括信息栏，高校搜索栏，指标指定栏，生成命令区域和文档预览区域。主页右上角有一个登出按钮在高校招生数据分析软件中保持常驻。当点击该按钮时前端会删除已获得的 token，并重定向至登录页面。



图 5.3 高校招生数据分析软件主页

主页选择高校界面可以通过两种方式选择，一种是直接通过输入高校全称搜索。另一种是通过搜索相关关键词联想出高校全部名称。其中，搜索功能如图 5.4

所示。

高校信息分析系统

退出

学校选择

学校

年份

西安电子科技

西安电子科技

西安电子科技大学

YYYY

☐ 否

是否为211

☐ 是

☒ 否

是否为双一流

☐ 是

☒ 否

确定

指标选择

☐ 招生专业（类）

☐ 按志愿录取率

☐ 生源指数

☐ 全国平均排名

☐ “千量”指数

☐ 学生信息指标

图 5.4 学校搜索功能

主页包含六大指标选项，对应了第四章设计与实现中提到的数据指标。通过选择可以指定生成文档是否包含特定内容。生成文档的过程较长，生成一个包含“招生专业（类）”、“全国平均排名”、“生源指数”的高校招生数据分析报告需要使用 76.44 秒，测试结果如图 5.5 所示。

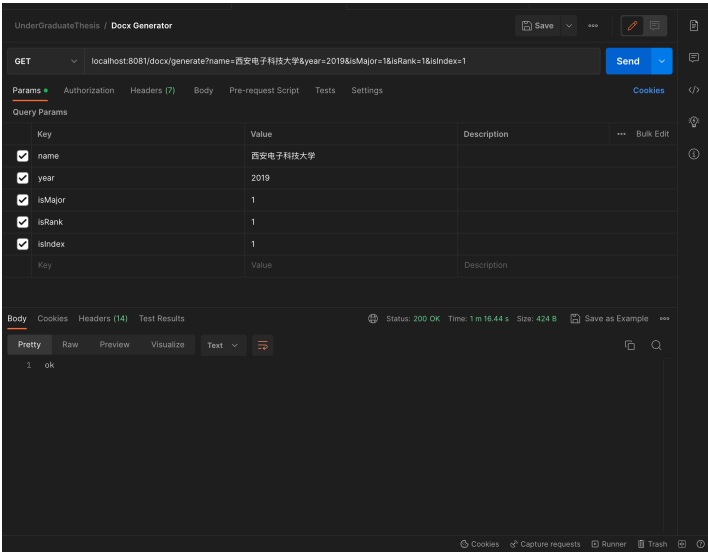


图 5.5 文档生成测试结果

为了使得高校数据招生软件在生成文档时有良好的视觉体验，在生成文档时，前端会播放生成加载动画。加载动画的动画效果如图 5.6 所示。当加载动画播放完毕后，前端会接收文档数据并下载至本地，下载地址默认为设别的下载文件夹对于我的测试设备，下载路径为” /Downloads”。

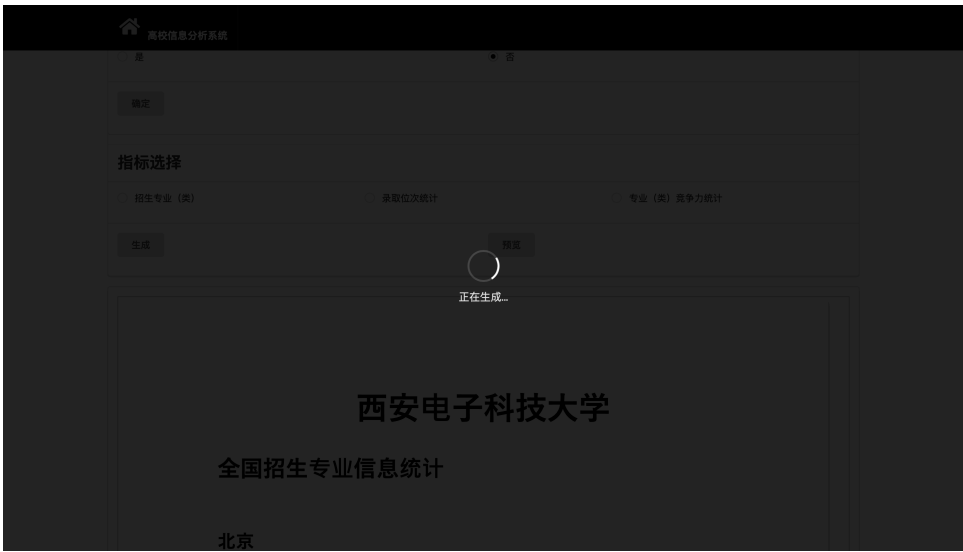


图 5.6 生成动画页面

点击主页中的预览按钮，文档将以 PDF 的方式在浏览器端渲染供用户预览。由于 PDF 和.docx 文档间存在部分标准差异，因此预览效果和实际文档效果有微小差别。预览效果如图 5.7 所示。



图 5.7 预览页面

此外，笔者也设计了针对单个模块功能的功能测试。以测试单个模块功能是否能够实现。对于文档生成模块，设计了一个测试程序测试各种格式的文档能否生成，包括标题，一级标题，二级标题，表格，图片，引用等。经测试，这些功能均可正常使用。测试结果如图 5.8 所示。

对于指标模块，设计了相应的测试。由于数据涉及到个人隐私，所以测试时所使用的数据为自动生成的数据。测试的分析指标有：录取人数统计（图 5.9），分专业录取情况（图 5.10），性别比例分布（图 5.11），专业录取比例（图 5.12）等。

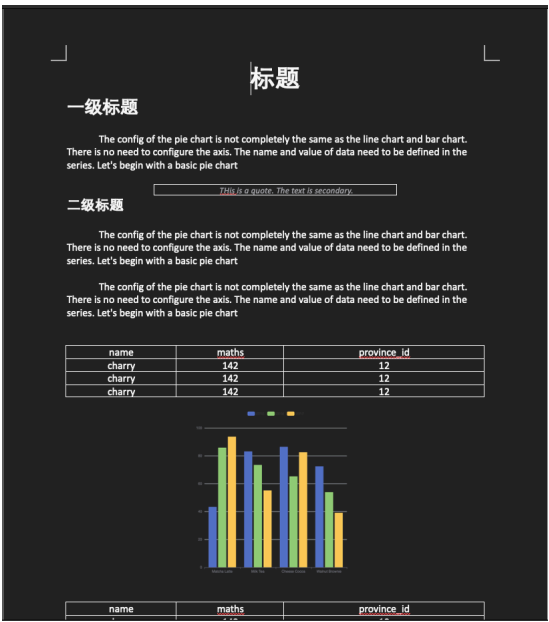


图 5.8 预览页面

经过图表数据与数据库数据计算结果进行比对，可以认为数据指标模块功能正常。测试结果如下列图所示。

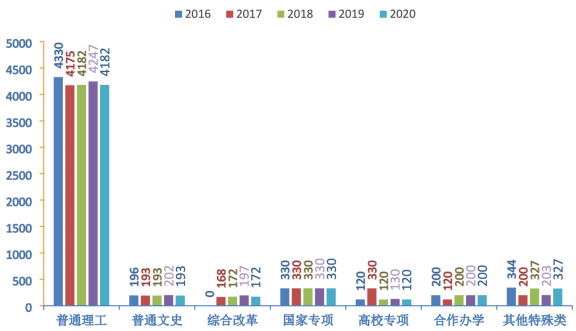


图 1 2016-2020 本科各类型录取人数情况

图 5.9 高校录取人数统计

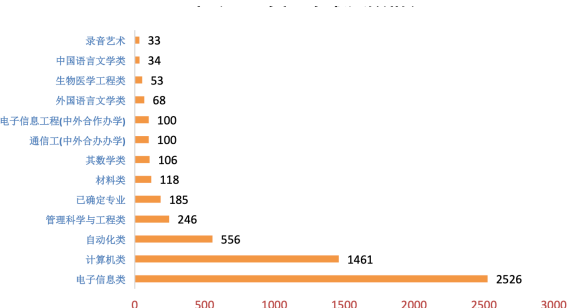


图 2 2022 年本科专业(类)录取情况

图 5.10 分专业录取情况

此外，笔者也设计了对大规模语言模型模块的测试。由于数据涉及到个人隐私，所以测试时所使用的提示为使用自动生成数据生成的段落。笔者设计了十组

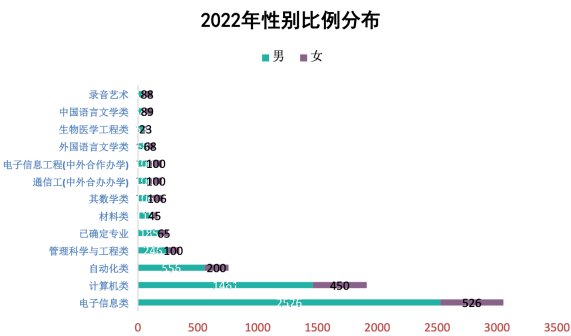


图 4 2022 年本科新生性别比例

图 5.11 高校学生性别比例分布

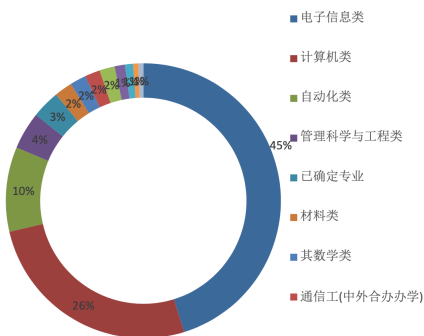


图 3 2022 年本科专业（类）录取情况

图 5.12 高校专业录取比例

不同的测试用例，测试大规模语言模型模块的生成文档的效果。经过测试，可以认为大规模语言模型模块功能正常。下面是其中使用到的一组用例。

- 1. **原始输入** 帮我生动的描绘” 山西省共有 203244 名同学报考。成绩分布见上图。如图所示，语文最高分 130，最低分 99，平均分 115，最高分考生姓名为：井聪健...”

**GPT 3.5 Turbo 生成结果** 在今年的高校招生中，山西省共有 203244 名同学报考，这是一个庞大的数字。对于这样的人数中，我们需要了解他们的考试情况，最好的方式就是看看考试成绩的分布情况。据上图所示，在语文考试中，最高分为 130 分，最低分为 99 分，平均分为 115 分，这是一个具有挑战性和竞争性的科目。

在这个大数据中，有一位同学名叫井聪健，他获得了语文科目最高分，这是一个非常令人惊叹的成就。相信这位同学在平日里一定非常用心，认真地学习，才能在考试中取得如此出色的成绩。

总的来说，这次高校招生中，山西省的同学们展现出了强大的学习实力和扎实的基础。他们的努力和拼搏在这份数据中得以体现，相信他

们会在未来的学习和生活中不断成长和进步。

2. **原始输入** 西安电子科技大学计算机专业的“全国平均排名”分别是: 2017 年 8575.43105 2018 年 6666.74175714 2019 年 6375.215275 2020 年 4595.42857333 2021 年 4841.93817097。

**GPT 3.5 Turbo 生成结果** 通过对西安电子科技大学计算机专业的“全国平均排名”指数的历年数据分析,我们可以发现以下趋势和结论。首先,从 2017 年到 2021 年,“全国平均排名”指数呈现逐年下降的趋势,这说明了西安电子科技大学计算机专业的排名不断地提升,越来越受到社会和学生的认可和追捧。其次,我们可以注意到,从 2017 年到 2020 年,“全国平均排名”指数下降的速度比较缓慢,但是在 2021 年,下降的速度明显加快,这说明了西安电子科技大学计算机专业在疫情影响下,迅速适应网上教学的新形势,优化教育教学模式,加强师资队伍建设和科学研究,提升了专业实力和社会声誉。最后,透过这些数据,我们可以看出西安电子科技大学计算机专业的素质在不断提高,同时说明了该专业在全国拥有越来越高的声望和认可度,这是该学科做出的巨大贡献和不断创新的结果,也是社会对该专业优秀教师和学生的认可和肯定。

3. **原始输入** 西安电子科技大学计算机专业的“千量指数”分别是: 2017 年 5.3491379 2018 年 8.033 2019 年 6.9612778 2020 年 6.461 2021 年 8.148。

**GPT 3.5 Turbo 生成结果** 这里有一组数据,它是关于西安电子科技大学计算机专业的“千量指数”值。这个指数是一个可以用来分析该专业的招生难易程度和稳定性的指标。我们可以看到,2018 年的“千量指数”值是最高的,高达 8.033,而 2017 年的“千量指数”值只有 5.3491379,最低,而 2021 年的值也非常高,高达 8.148。根据这些数据,我们可以得出一些结论。首先,从最高值来看,2018 年是计算机专业的招生难度最大的一年。相比之下,2017 年的招生表现较好,招生额更为稳定。其次,2019 年和 2020 年的“千量指数”值都比 2018 年低,说明这两年计算机专业的招生稳定程度也比 2018 年更好。总的来说,这组数据表明,计算机专业的招生情况不断发展变化,但从“千量指数”值上看,招生难度和稳定程度都比较高,这也说明了该专业的竞争激烈程度。

## 5.2 软件性能分析

针对高校招生数据分析软件进行性能分析，发现高校招生数据分析软件的性能占用主要集中在以下几方面：

1. 数据库占用：高校招生数据分析软件需要频繁读取数据库中数据；
2. 服务器端占用：服务器端需要处理客户端请求，进行数据指标计算；
3. 文档生成和转换占用：服务器端文档生成和格式转换需要使用计算资源；
4. 浏览器端占用：浏览器端需要进行渲染和路由工作；

通过软件开发过程中进行的过程测试，可以得到结论，数据库读取、服务器端的数据指标计算，和浏览器端的渲染任务使用主要的计算资源。大规模语言模型模块使用主要的互联网带宽资源，因为大规模语言模型辅助模块需要接入互联网运行。经过软件性能分析，可以认为高校招生数据分析系统可以在绝大部分服务器和其他设备上运行。

## 5.3 软件性能测试

针对高校招生数据软件，笔者设计了软件性能测试流程并对其进行测试。该流程基于两种性能的设备进行，一种性能较为充裕，一种性能为普通性能。在流程上，主要包括以下几个方面的测试：

1. 数据库读取测试：按照数据查询模块的 API 设计多次重复性读取测试，测试读取时间和内存占用；
2. 服务器端测试：按照数据指标计算模块的 API 设计多次重复性读取测试，测试计算时间和此时服务器端内存占用；
3. 文档生成和转换占用：按照标准格式文档设计多次重复性生成和格式转换测试，记录用时和内存占用；
4. 浏览器端占用：记录浏览器端渲染相应界面的时间和内存占用。

软件性能测试基于两种性能设备进行。由于笔者只有一台设备，但由于笔者设备架构的特殊性。笔者设计了一种测试方法，可在一种设备上模拟两种设备性能的测试方案。笔者的设备具体配置如表 5.1 所示。由于笔者设备的 CPU 基于 ARM 架构，与目标客户的服务器和其他设备的 CPU 架构不同。而高校招生数据软件使用到的技术和软件全部具有针对 Apple M1 芯片的优化。此外，Apple M1 可以在损耗一定性能下通过 Rosetta 2 运行基于 x86 架构的软件。因此，笔者通过在设备上经过优化的软件编译出的高校招生数据软件和未经过优化的软件编译出的版本进行比较，两种版本的运行效率可以认为是两种不同性能设备的运行效率。



表 5.1 设备配置

部件	型号
CPU	Apple M1 chip 8-core
GPU	8-core integrated GPU
RAM	16GB unified memory
ROM	2TB SSD
OS	macOS Ventura 13.3.1
Arch	ARM 64

CPU 占用率反应出对计算资源的需求量。经过测试，可以认为计算资源主要消耗在服务器端数据指标计算上。在小型测试全速运行时，大约会占用 1.2% 的 CPU 计算资源。约占整体项目使用计算资源的 66.7%。

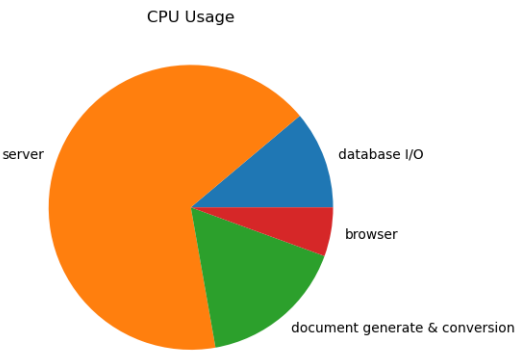


图 5.13 CPU 占用率

内存占用率反应出对内存资源的需求量。经过测试，可以认为内存资源主要消耗在数据库读取上。这主要是在测试过程中，模拟了软件实际部署运行时多次对数据库读取会在内存中存放大量临时性数据。在小型测试全速运行时，大约会占用 520.9MB 的内存资源。约占整体项目使用内存资源的 40.9%。

在运行未经优化编译程序和优化编译程序测试时，软件的 RAM 占用率未有明显变化。这主要是因为没有计入 Rosetta 2 带来的额外 RAM 占用。实际在性能较差的 x86 设备运行该软件时并无该 RAM 占用。且 CPU 占用相对比例也无变化。但 CPU 总体占用提高约 20%。运行时间相比经过优化版本也多出了约 23%。经过优化版本运行单次测试需要 5.34 min，而未优化版本软件运行单词测试需要 6.58 min。

通过软件性能测试，可以认为高校招生数据软件基本可以在绝大多数主流服务器和设备上部署运行。

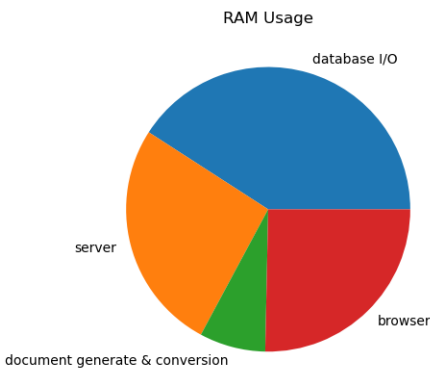


图 5.14 RAM 占用率

### 5.4 本章小结

本章对高校招生数据分析软件进行了软件性能分析，并根据软件性能分析设计了软件性能测试流程。通过笔者设备特性巧妙地实现了在一种设备上模拟两种设备性能测试的方案。最后给出了性能测试结果，得出软件可以在绝大多数终端部署运行的结论。

## 第六章 结论

### 6.1 软件优点与亮点

高校数据分析软件主要有四个优点。第一，高校招生数据分析软件将原本的高校招生数据进行了预处理工作，降低了数据的存储占用。第二，减少了人力资源的使用。将原本需要人工完成的数据分析工作交给了计算机完成。第三，高校招生数据分析软件内置了一些数据分析指标计算模块，方便根据数据得到更加准确的结论。第四，可以将分析数据汇总并结合数据可视化的方法使用软件自动生成格式规范的.docx 文档。

高校数据分析软件主要有两个亮点。第一，高校数据分析软件创造性地引入了大规模语言模型技术辅助生成个性化文档，取得了较好的效果。大规模语言模型技术不仅可以生成更加生动的文字，还有相当大概率生成一些数据指标无法反映的结论。第二，高校数据分析软件通过 Java 实现.docx 到.pdf 的文件转换，将.pdf 渲染至浏览器端实现用户预览。这样做的好处是不需要客户在浏览器端安装任何插件即可将生成的.docx 文档进行预览。

### 6.2 软件局限性

高校数据分析软件主要有三个局限性。第一，由于全国近年来实行的高考改革，导致不同年份之间的数据格式不同，高校招生数据分析软件对不同格式的数据的支持还不够友好。第二，由于大规模语言模型技术生成文本具有一定程度的不稳定性，以及网络连接的不稳定性，高校招生数据分析软件无法保证生成的个性化报告文本完全客观准确。第三，高校招生数据分析软件在预览部分只能实现在线预览而无法进行在线修改。

### 6.3 软件可优化方向

针对目前高校招生数据分析软件的局限性，笔者认为该软件存在几个可优化方向。第一，可以研究对不同格式的高校招生数据的统一处理方法，使得高校招生数据分析软件可以更加有效地分析高考改革前后的高校招生工作。第二，通过构建提示词等方法稳定大规模语言模型的输出表现，或者在收集到足够量数据后通过这些数据微调成熟的大规模语言模型如 LLaMA 等，实现可部署在本地的足够稳定的应用于高校招生分析工作的大规模语言模型。第三，通过引入在线编辑

器等方法，或者自己实现 Microsoft Word 文件到浏览器端的映射方法，实现在线修改生成的报告文档，提高工作效率。

## 致谢

当毕业论文写到致谢部分的时候，我才意识到大学看似漫长的四年到现在只能以倒计时计。在大学四年生活的最后关头，我想对很多人说声感谢。感谢他们让我的大学生活丰富多彩，让我在大学度过了快乐的时光。

感谢我的家人，爸爸妈妈和爷爷奶奶，他们对我的支持至关重要，我永远都会将亲情放在第一位。

感谢我的毕业论文导师谢琨教授，她在毕业论文方面提供了很多宝贵的意见和建议，让我的论文更加规范。谢琨老师是我遇到的大学中最棒的老师，她的离散讲的真的很好。

感谢大学四年期间结识的朋友，尤其是大鸟转转的各位群友。我们不仅经常一起娱乐，而且我们之间可以进行很多深层次问题的探讨。虽然这很像魏晋时期的“清谈”，但还是丰富了我的人生观和价值观。可以说认识他们是我大学四年最棒的事。他们认真的学习态度和开阔的学术视野鞭策我继续朝着优秀的方向努力。大鸟转转的诸君马上就要各奔东西去下一站更广阔的天地闯荡，在这里只能祝福诸君一帆风顺，前程似锦。友谊万岁！

感谢大学四年的室友秦江、王帅博等。他们让我在宿舍中也不至于那么无聊，虽然我平日在宿舍的时间并不长。

感谢大学四年还一直保持联系的杨萌同学。西安到苏州的距离没有阻隔我们的联系。感谢她能听我的唠叨，帮助我在困难的时期走出低谷。祝愿她能够在新的研究生阶段获得更大的学术成功。

在大学的最后，我由于在实习公司中见到的一些事情思想从就业转为了出国留学。在自己的不懈努力下，在半年不到的时间内，我从六级不过的英语小白达到了目标院校的 TOEFL 和 GRE 要求，也丰富了我的简历。十分感谢谢琨教授、魏静萱副教授和王凯东副教授在我的留学申请中提供推荐信。

幸运的是，我收到了来自目标院校的录取通知。今年八月，我就要踏上加利福尼亚州的土地，开始自己的留学生活。希望在未来的几年中能够继续保持大学期间的乐观、努力。也希望能和大学期间的朋友们尤其是大鸟转转的各位群友们保持联系，共同进步。



## 参考文献

- [1] 中华人民共和国国家统计局, “中华人民共和国 2022 年国民经济和社会发展统计公报,” 国家统计局, 政府公报, 2022.
- [2] 乔琳琳, “我国高校信息化发展状况综述,” 东北电力大学, Tech. Rep., 06 2022.
- [3] Y. Zhuang and Z. Gan, “A machine learning approach to enrollment prediction in chicago public school,” in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, 2017, pp. 194–198.
- [4] E. Begoli and Edmon, “A short survey on the state of the art in architectures and platforms for large scale data analysis and knowledge discovery from data,” 08 2012.
- [5] L. Guo, “Research on data analysis and mining technology based on computer visualization,” 10 2020, pp. 194–200.
- [6] Z. Li, L. Wang, W. Piao, and L. Zhang, “Design and implementation of production visual report subsystem based on echarts,” in *2022 IEEE Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, 2022, pp. 696–701.
- [7] T. R. Ruiz and G. Nuñez, “Characterization of enrollment in higher education in ecuador,” in *2022 IEEE Sixth Ecuador Technical Chapters Meeting (ETCM)*, 2022, pp. 1–4.
- [8] W. Mckinney, “pandas: a foundational python library for data analysis and statistics,” *Python High Performance Science Computer*, 01 2011.
- [9] M. Bauer and M. Garland, “Legate numpy: Accelerated and distributed array computing,” in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, ser. SC '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: <https://doi.org/10.1145/3295500.3356175>
- [10] S. van der Walt, S. Colbert, and G. Varoquaux, “The numpy array: A structure for efficient numerical computation,” *Computing in Science and Engineering*, vol. 13, pp. 22–30, 05 2011.
- [11] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, and J.-R. Wen, “A survey of large language models,” *arXiv preprint arXiv:2303.18223*, 2023. [Online]. Available: <http://arxiv.org/abs/2303.18223>
- [12] C. Dong, Y. Li, H. Gong, M. Chen, J. Li, Y. Shen, and M. Yang, “A survey of natural language generation,” 12 2021.
- [13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 06 2017.
- [14] M. Ganesh, A. Raj, and I. T. J. Swamidason, “An analysis of the significance of spring boot in the market,” 07 2022, pp. 1277–1281.

- 
- [15] L. xuchen and L. chaoyu, “Design and implementation of a spring boot-based data collection system,” in *2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 1, 2020, pp. 236–239.
  - [16] M. Islam, S. Rahaman, N. Meng, B. Hassanshahi, P. Krishnan, and D. D. Yao, “Coding practices and recommendations of spring security for enterprise applications,” in *2020 IEEE Secure Development (SecDev)*, 2020, pp. 49–57.
  - [17] D. Ltd., “2021 spring framework user survey,” Diffblue Ltd., Survey, 12 2021.
  - [18] C. M. Novac, O. C. Novac, R. M. Sferle, M. I. Gordan, G. BUJDOSó, and C. M. Dindelegan, “Comparative study of some applications made in the vue.js and react.js frameworks,” in *2021 16th International Conference on Engineering of Modern Electric Systems (EMES)*, 2021, pp. 1–4.
  - [19] X. HongJu, W. Fei, W. FenMei, and W. XiuZhen, “Some key problems of data management in army data engineering based on big data,” in *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*, 2017, pp. 149–152.
  - [20] J. J. J. Sarjeant, “The axios http client for the browser and node.js,” The Axios Project, Tech. Rep., 2022.