

中国大学生计算机设计大赛

**考研择校辅助平台——考研帮**

软件应用与开发

**作品大类：**

**作品小类：** Web 应用与开发

**作品编号：** 2024022085

**参赛成员：**

丁锐锐，韩韬，胡嘉辉，李逸冉，贾径懿

**指导教师：**

马明

**完成时间：** 2024 年 5 月 6 日

目录

**[第一章](#bookmark2)****[需求分析](#bookmark2)****[4](#bookmark2)**

[1. 项目背景 4](#bookmark4)

[2. 项目立意 4](#bookmark6)

[3. 功能分析 4](#bookmark8)

**[第二章](#bookmark10)****[概要设计](#bookmark10)****[5](#bookmark10)**

[1. 功能模块 5](#bookmark12)

[2. 架构设计 5](#bookmark14)

**[第三章](#bookmark16)****[详细设计](#bookmark16)****[6](#bookmark16)**

[1. 界面设计 6](#bookmark18)

[2. 数据库设计 9](#bookmark20)

[3. 关键算法设计 11](#bookmark22)

**[第四章](#bookmark24)****[测试报告](#bookmark24)****[19](#bookmark24)**

[1. 目的 19](#bookmark26)

[2. 测试概述 19](#bookmark28)

[3. 测试结果 20](#bookmark30)

**[第五章](#bookmark32)****[安装及使用](#bookmark32)****[21](#bookmark32)**

[1. 环境要求 21](#bookmark34)

[2. 部署流程 22](#bookmark36)

[3. 典型使用流程 22](#bookmark38)

**[第六章](#bookmark40)****[项目总结](#bookmark40)****[22](#bookmark40)**

[1. 项目协调与任务分解 22](#bookmark42)

[2. 项目总结 22](#bookmark44)

[3. 后续改进思路 23](#bookmark46)

**[第七章](#bookmark48)****[参考文献](#bookmark48)****[23](#bookmark48)**

**图目录**

[图 2.1 功能模块图 5](#bookmark50)

[图 2.2 架构设计图 6](#bookmark52)

[图 3.1 通用界面展示 6](#bookmark54)

[图 3.2 院校库（1） 7](#bookmark56)

[图 3.3 院校库（2） 7](#bookmark58)

[图 3.4 院校库（3） 7](#bookmark60)

[图 3.5 专业库 8](#bookmark62)

[图 3.6 AI 择校（1） 8](#bookmark64)

[图 3.7 AI 择校（2） 9](#bookmark66)

[图 3.8 ER 图 10](#bookmark68)

[图 3.9 志愿推荐流程 11](#bookmark70)

[图 3.11 召回流程图 12](#bookmark72)

[图 3.12 FAST TEST 模型结构图 12](#bookmark74)

[图 3.13 LSTM 与 XGBOOST 训练集 LOSS 曲线 15](#bookmark76)

[图 3.14 录取概率预测模型 16](#bookmark78)

**表目录**

[表 3.1 院校表 9](#bookmark80)

[表 3.2 专业表 9](#bookmark82)

[表 3.3 专业信息表 10](#bookmark84)

[表 3.4 学院表 10](#bookmark86)

[表 3.5 考试科目表 10](#bookmark88)

[表 3.6 部分电子信息学科二级目录 11](#bookmark90)

[表 3.7 专业标签预测情况 13](#bookmark92)

[表 3.8 通过 FASTTEXT 推荐与标签一类专业及院校 13](#bookmark94)

[表 3.9 部分样本特征得分状态以及默认权重 14](#bookmark96)

[表 3.10 推荐专业分类等级 17](#bookmark98)

[表 3.11 数据来源 18](#bookmark100)

[表 4.1 功能测试结果 20](#bookmark102)

[表 4.2 性能测试结果 20](#bookmark104)

[表 4.3 安全测试结果 21](#bookmark106)

[表 4.4 兼容性全测试结果 21](#bookmark108)

[表 5.1 环境要求 22](#bookmark110)

[表 6.1 任务分解与项目协调 22](#bookmark112)

**第一章** 需求分析

**1. 项目背景**

当前，考研报名人数持续攀升，但与此同时，也面临着录取率逐年下降的现实。 这既反映了高等教育资源的相对稀缺性，也体现了考生间激烈的竞争态势。在这种背景下，考研不仅仅是对知识的考验，更是对信息收集能力的检验。随着考研热度的持续升温，越来越多的学子在考研过程中面临着一个重要的选择——择校。

**2. 项目立意**

择校不仅关乎到未来几年的学习环境和学术氛围，更直接影响到个人的职业发展 道路。然而，由于信息不对称和缺乏专业的指导，许多考生在择校过程中感到迷茫和困惑。我们的平台面向广大纠结与考验择校的考生，考生获取考研信息的地方更多是 中国研究生招生信息网和各大学校官网，以及各大论坛。但这些地方存在消息零散、 难收集、不能横向对比等缺点。因此，我们开发了这款考研择校辅助网站，旨在为广大考研学子提供一个全面、准确、便捷的择校平台。

**3. 功能分析**

**3.1 院校库**

在院校库中，用户可以查看全国的院校，可以通过所在省份和学校层次（985,211,其他）来筛选心仪的院校，也可以通过模糊搜索来寻找指定的院校。选择院校后可以查看该院校开设的专业，选择该院校开始的某个专业后，可以查看最近一年的录取名单，最近一年的分数线，和连续三年的招生情况，同时还可以查看该院校该专业的考试科目。

**3.2** **专业库**

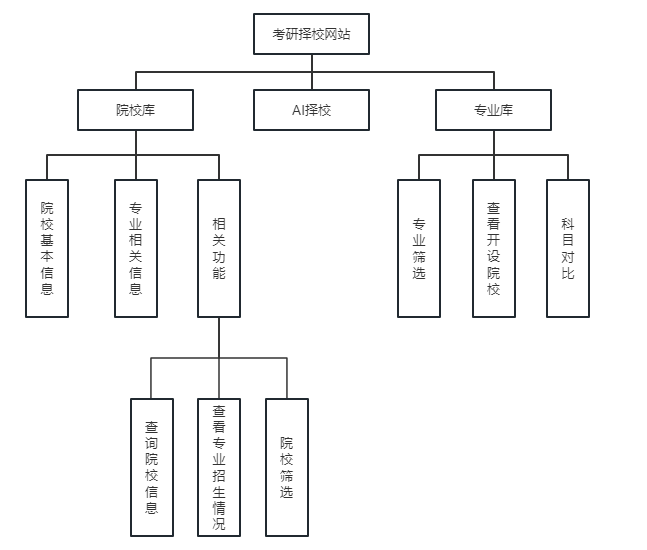
在专业库中，用户选择一个专业后，可以查看开设的所有院校，也可根据省份和学校层次筛选开设该专业的院校。选中两个学校后可以横向对比两个学校的分数线和 招生人数。

**3.3 AI择校**

用户输入目标专业、预估分数、目标省份和期望的学校层次后，模型会返回适合的院校列表供用户选择。

**第二章** 概要设计

**1. 功能模块**



**图** **2.1** **功能模块图**

**2.** **架构设计**

2.1 **表示层：**

使用 Vue 3 框架构建交互式 Web 界面，配合 Axios 实现 AJAX 异步 通信，发送 HTTP 请求。

2.2 **接入层**：

采用 Nginx 服务，负责路由分发，确保后端服务的负载均衡。

2.3 **服务层**：

基于 Go 语言构建的应用服务，通过 Docker 容器进行多机部署，确 保服务的高可用性。

使用轻量级的 Gin 框架作为 HTTP Web 框架，数据访问层采用 GORM 框架。

2.4 **数据层**：

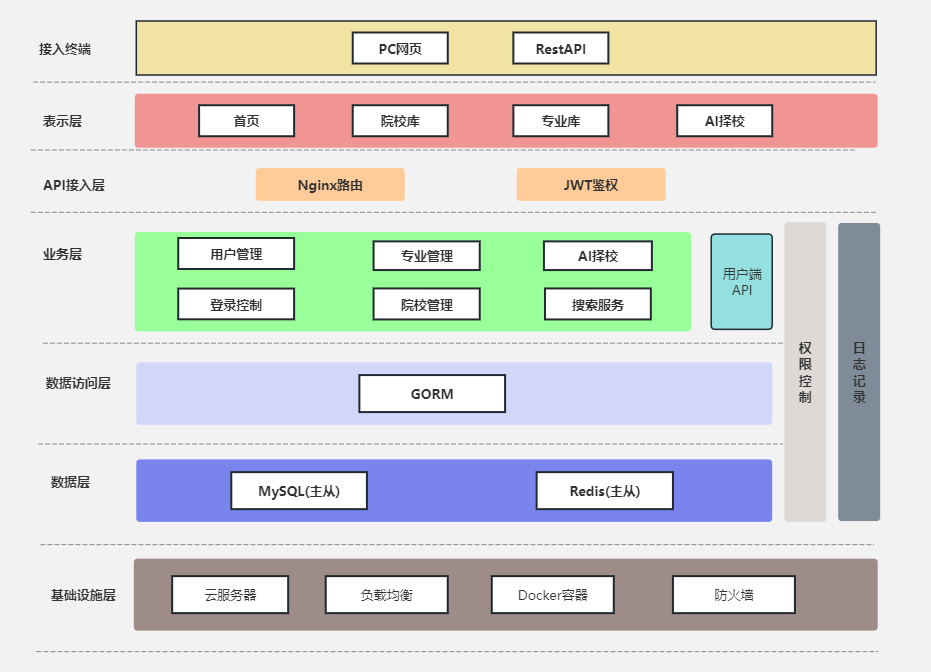
MySQL 采用主从复制模式，支持读写分离，提升数据库的性能与可用 性。

Redis 也采用主从模式，作为旁路缓存，提高系统整体性能和响应速 度。

2.5 **存储策略**：

MySQL 作为主数据库，用于保存全量数据。

Redis 作为缓存数据库，优化数据访问速度，减轻主数据库负载。



**图** **2.2** **架构设计图**

**第三章** 详细设计

**1.** **界面设计**

**1.1 通用界面**

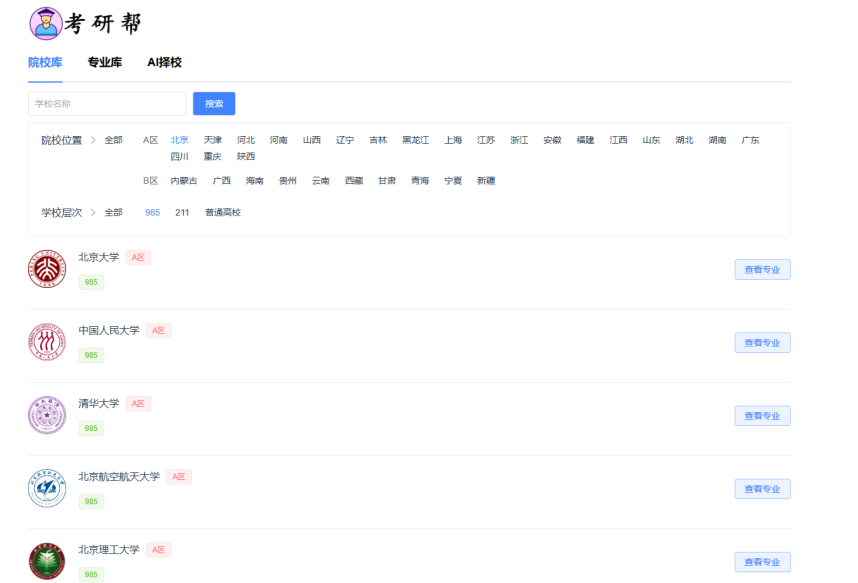
在每个页面上方有导航栏，显示 LOGO，同时提供了三个切换按钮。



**图** **3.1** **通用界面展示**

**1.2 院校库**

显示全国的院校列表，在上方有模糊搜索框和组合筛选框，可以通过院校位置和学校 层次筛选院校，在下方显示对应的院校。



**图 3.2 院校库（1）**

点击查看专业后可以查看某学校开设的专业，以列表的形式显示每个专业的学院名称，专业名称，专业代码，培养方式。同时支持搜索某一具体专业的功能。



**图** **3.3** **院校库（2）**

点击查看后可以查看该院校该专业的最近一年录取名单，分数分析，近几年招生人 数，考试范围，考生分析。



**图** **3.4** **院校库（3）**

**1.3 专业库**

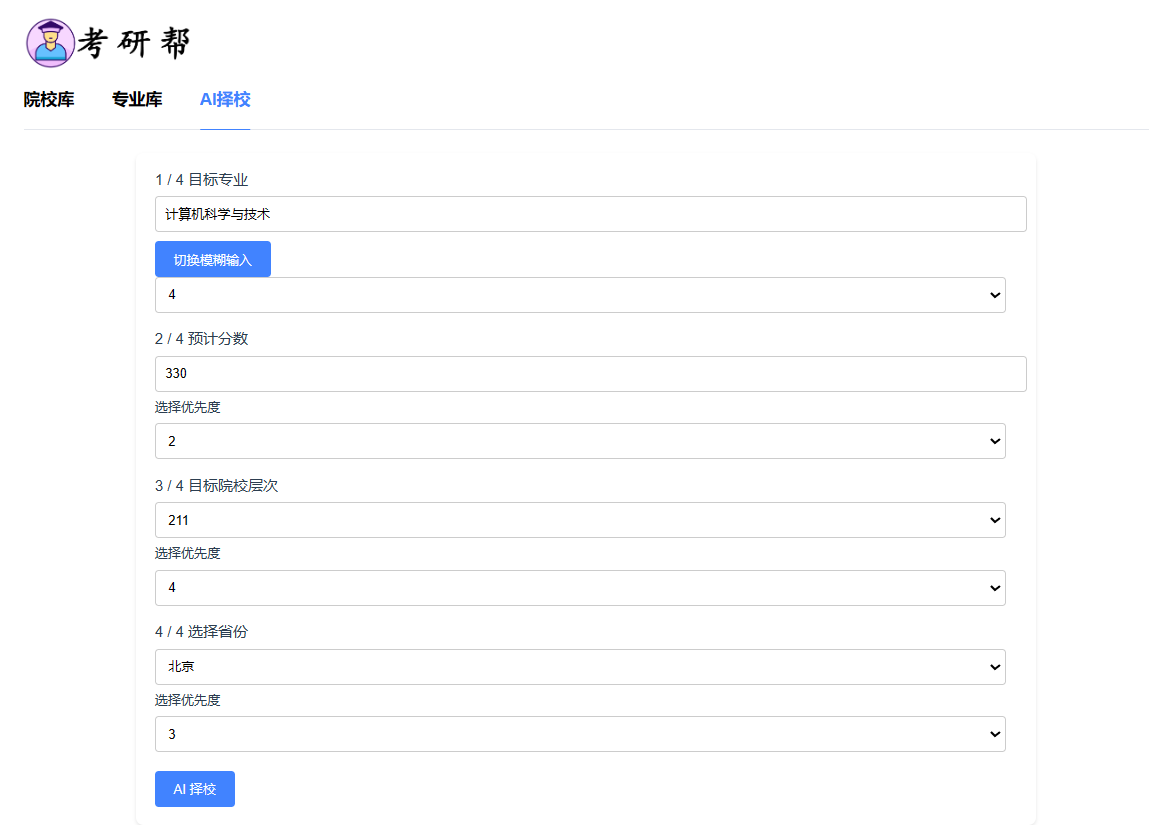
通过学科门类和学科类别筛选专业，点击开设院校按钮即可查看开设该专业的所有院 校。



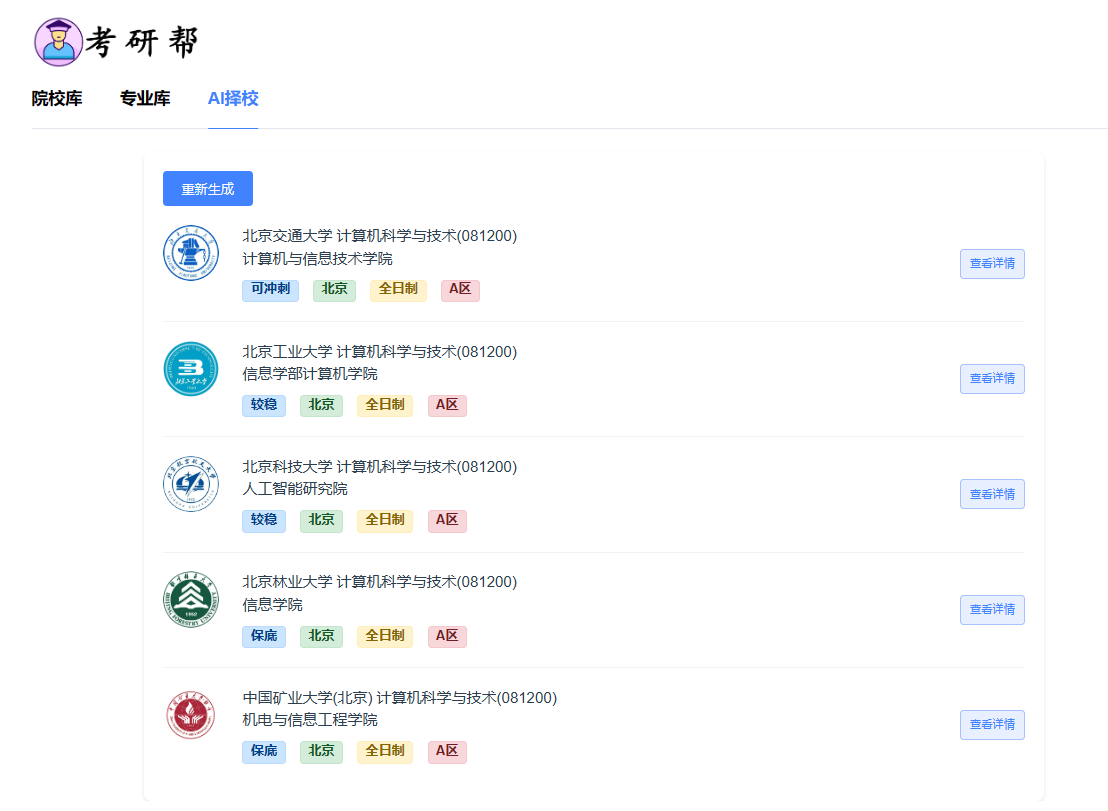
**图** **3.5 专业库**

**1.4 AI 择校页面**

输入目标专业、目标分数，目标省份和目标院校层次，同时调整优先级，点击 AI择校会生成可选择的院校。



**图 3.6 AI 择校（1）**



**图 3.7 AI 择校（2）**

**2.** **数据库设计**

**2.1 院校表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **列名** | **类型** | **长度** | **不能为空** | **主键** | **注释** |
| id | bigint | 32 | 否 | 是 | 主键 |
| university\_name | varchar | 20 | 否 | 否 | 大学名称 |
| location | varchar | 255 | 是 | 否 | 所在地 |
| school\_level | varchar | 255 | 是 | 否 | 学校层次 |
| logo | varchar | 255 | 是 | 否 | 校徽 |
| introduction | varchar | 255 | 是 | 否 | 简介 |
| updated\_at | datetime | 3 | 是 | 否 | 更新时间 |
| created\_at | datetime | 3 | 是 | 否 | 创建时间 |

**表** **3.1** **院校表**

**2.2** **专业表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **列名** | **类型** | **长度** | **不能为空** | **主键** | **注释** |
| id | bigint | 32 | 否 | 是 | 主键 |
| major\_name | varchar | 32 | 是 | 否 | 专业名称 |
| major\_code | varchar | 10 | 是 | 否 | 专业代码 |
| university\_id | bigint | 32 | 是 | 否 | 外键，指向大学表的主键 |
| college\_id | bigint | 32 | 是 | 否 | 外键，指向学院表的主键 |

**表** **3.2** **专业表**

**2.3** **专业信息表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **列名** | **类型** | **长度** | **不能为空** | **主键** | **注释** |
| year | varchar | 255 | 是 | 否 | 年份 |
| training\_methods | varchar | 255 | 是 | 否 | 培养方式 |
| min\_score | varchar | 255 | 是 | 否 | 最低分 |
| max\_score | varchar | 255 | 是 | 否 | 最高分 |
| average\_score | varchar | 255 | 是 | 否 | 平均分 |
| research\_direction | varchar | 255 | 是 | 否 | 研究方向 |
| total\_enrollment | bigint | 32 | 是 | 否 | 总招生人数 |
| first\_choice\_admissions | bigint | 32 | 是 | 否 | 第一志愿录取人数 |
| adjustment\_personnel | bigint | 32 | 是 | 否 | 调剂录取人数 |

**表** **3.3** **专业信息表**

**2.4** **学院表**

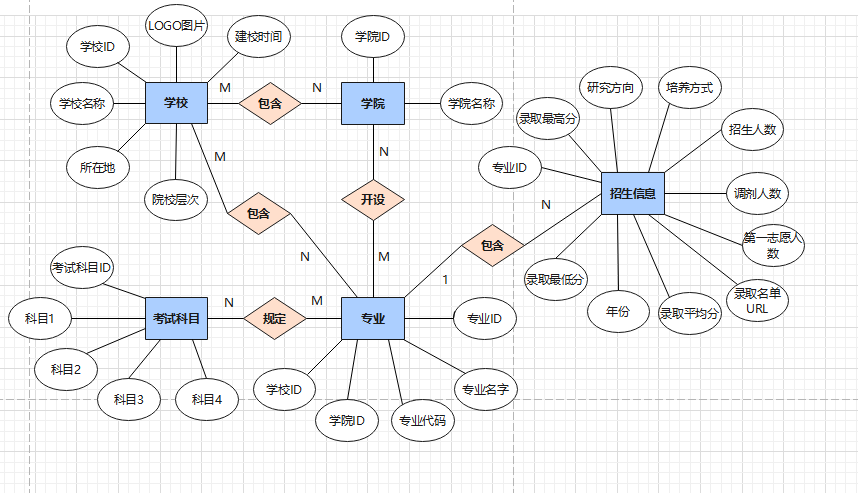
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **列名** | **类型** | **长度** | **不能为空** | **主键** | **注释** |
| college\_id | bigint | 32 | 否 | 是 | 主键 |
| college\_name | varchar | 255 | 是 | 否 | 学院名称 |
| university\_id | bigint | 32 | 是 | 否 | 外键，指向大学表 的主键 |
| created\_at | datetime | 3 | 是 | 否 | 创建时间 |
| updated\_at | datetime | 3 | 是 | 否 | 更新时间 |

**表** **3.4** **学院表**

**2.5** **考试科目表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **列名** | **类型** | **长度** | **不能为空** | **主键** | **注释** |
| id | bigint | 32 | 否 | 是 | 主键 |
| major\_id | bigint | 32 | 是 | 否 | 外键，指向专业表 的主键 |
| subject\_id | bigint | 32 | 是 | 否 | 科目 4 |
| replaceable\_subject\_ 1 | bigint | 32 | 是 | 否 | 科目 1 |
| replaceable\_subject\_2 | bigint | 32 | 是 | 否 | 科目 2 |
| replaceable\_subject\_3 | bigint | 32 | 是 | 否 | 科目 3 |

**表** **3.5** **考试科目表**

**图** **3.8** **ER** **图**

高考志愿推荐系统开发采用的数据由内蒙古易高教育科技有限公司，内蒙古大学学生职业发展联盟，内蒙古大学精英开发者联盟共同提供，一共包括 158269条数据。其中，本实验所需的关键数据集如下表所示。

|  |  |  |
| --- | --- | --- |
| **数据集** | **数据量** | **备注** |
| 往届考生数据集 U | 816 条 | 包含考生个人信息、心理学测试结果等数据 |
| 专业数据集 M | 1253 条 | 包含专业名称、专业概况等数据 |
| 专业属性数据集 A | 8269 条 | 包含薪酬水平、考研率等数据 |
| 院校数据集 S | 2756 条 | 包含院校名称、院校代码、开设专业等数据 |
| 院校录取分数线数 据集 L | 13725 条 | 各院校不同专业的近四年复试分数线数据 |

**表** **3.11 数据来源**

往届考生数据集 U：包括 816 条内蒙古自治区内高校往届考生数据，主要由往届考生报考专业、专业数据集 M：包括 1253 条专业数据，主要对专业进行描述，包含专业名称、专业代码和专业概况和专业特征等信息。

专业属性数据集 A：包括 8269 条专业属性数据，主要包含薪酬水平、深造率、考公率和就业率等专业属性信息。

院校数据集 S：包括 2756 条院校数据，主要包含院校名称、院校代码、开设专业和院校类型等信息。

院校录取分数线数据集 L：包括 13725 条数据，主要是各院校 2020-2023 不同专业的研究生复试分数线。

**第四章** 测试报告

包括测试报告和技术指标。为了保证作品质量，建议多进行测试，并将测试用例、测试过程、测试结果、修正过程或结果形成文档，也可以将本标题修改为主要测试，撰写主要测试过程结果及其修正识性内容。

**1.** **目的**

根据需求编写测试计划，在此基础上收集和设计测试用例，且通过对被测系统基本功能、操作流程和业务流程的熟悉，执行测试用例, 找出该系统在功能、界面、易用性等方面的缺陷, 提交缺陷报告。目的在于总结测试阶段的测试以及分析测试结果, 描述系统是否符合需求。

**2.** **测试概述**

**2.1** **测试范围**

本次测试主要针对系统 V1.0 版本的以下模块进行了功能测试、性能测试、安全测试和兼容性测试：

学校信息管理

专业信息管理

考试科目管理

学校筛选

联合查询

**2.2** **测试方法**

本次测试采用了以下方法：

**功能测试**：使用黑盒测试方法，根据需求规格说明书编写并执行了功能测试用例，验证软件是否满足用户需求。

**性能测试**：使用负载测试方法，模拟多个用户同时访问系统，监测并分析系统的响应时间、吞吐量、资源利用率等性能指标。

**安全测试**：使用渗透测试方法，尝试利用系统存在的漏洞进行攻击，检测并评估系统的安全性。

**兼容性测试**：使用多浏览器、多操作系统、多分辨率等不同环境条件下进行测试，检测并评估系统的兼容性。

**2.3** **测试环境**

本次测试使用了以下环境：

测试服务器：Ubuntu Server 22.04 LTS 64bit, CPU - 2 核, 内存 - 2GB IP: <8.141.12.156>

数据库服务器：Ubuntu Server 22.04 LTS 64bit, CPU - 2 核, 内存 - 2GB IP: <8.141.12.156>, MySQL – 5.7.40

应用服务器：Ubuntu Server 22.04 LTS 64bit, CPU - 2 核, 内存 - 2GB IP: <8.141.12.156>

测试客户端：Windows 11, CPU: - AMD Ryzen 7 5800H with Radeon Graphics

3.20 GHz, RAM: - 16.0 GB, Chrome - 124.0.6367.158, Firefox – 125.0.3

测试工具：Apifox（测试管理工具），wrk（性能测试工具）

**2.4** **测试时间**

本次测试的时间安排如下

功能测试：2023 年 3 月 15 日-2023 年 4 月 1 日

性能测试：2023 年 4 月 1 日-2023 年 4 月 10 日

安全测试：2023 年 4 月 11 日-2023 年 4 月 20 日

兼容性测试：2023年 4 月 21 日-2023 年 4 月 28 日

**3.** **测试结果**

**3.1** **功能测试结果**

功能测试共编写了 120 个测试用例，执行了 120 个，通过了 113 个，未通过的 17 个均已提交缺陷报告。功能测试的覆盖率为 100%，通过率为 94.16%。功能测试的结果如下表所示：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 模块名称 | 测试用例数 | 执行用 例数 | 通过用 例数 | 未通过 用例数 | 缺陷数 | 覆盖率 | 通过率 |
| 学校信息管理 | 20 | 20 | 20 | 0 | 0 | 100% | 100% |
| 专业信息管理 | 20 | 20 | 20 | 0 | 0 | 100% | 100% |
| 考试科目管理 | 10 | 10 | 8 | 2 | 2 | 100% | 80% |
| 学校筛选 | 40 | 40 | 30 | 10 | 10 | 100% | 75% |
| 联合查询 | 30 | 30 | 25 | 5 | 5 | 100% | 83% |
| 总计 | 120 | 120 | 113 | 17 | 17 | 100% | 94.16% |

**表** **4.1** **功能测试结果**

**3.2** **性能测试结果**

性能测试共执行了 5 个性能测试用例，分别模拟了 10、20、50、100、200 个用户同时访问系统，并监测了系统的响应时间、每秒接口请求数、90%响应时间等性能指标。性能测试的结果如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 并发用户数 | 平均响应时间(ms) | 每秒接口请求 数 | 90%响应时间 | 总请求数 |
| 10 | 23 | 8.67 | 26 | 577 |
| 20 | 24 | 13.45 | 27 | 1716 |
| 50 | 25 | 16.05 | 28 | 4907 |
| 100 | 45 | 28.16 | 49 | 1901 |
| 200 | 150 | 28.20 | 456 | 1874 |

**表** **4.2** **性能测试结果**

**3.3** **安全测试结果**

安全测试共执行了 5 个安全测试用例，分别针对系统的身份认证、数据加密、输入验证、错误处理和日志记录等方面进行了渗透测试，检测并评估了系统的安全性。安全测试的结果如下表所示：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例编号 | 测试用例描述 | 测试结果 | 缺陷描述 | 缺陷等级 |
| TC01 | 尝试使用不合法的 用户名和密码登陆 系统 | 失败 | 系统拒绝了不合法 的登录请求 | 无 |
| TC02 | 尝试直接访问网站 首页绕过登录系统 | 失败 | 系统拦截了不合法 的请求, 需要用户验 证 | 无 |
| TC03 | 尝试使用 SQL 注入 攻击获取数据库的 敏感数据 | 失败 | 系统对 SQL 语句进 行了预编译, 没有出 现安全漏洞 | 无 |
| TC04 | 尝试访问系统中的 错误信息和日志文 件 | 成功 | 系统没有对错误信 息和日志文件进行 有效的隐藏和保  护，允许攻击者获 取系统的内部信息 和状态 | 中 |

**表** **4.3** **安全测试结果**

**3.4** **兼容性全测试结果**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 测试用例描述 | 测试结果 | 缺陷描述 | 缺陷等级 |
| TC01 | 用 Chrome 浏览器访 问网站 | 通过 | 系统在 Chrome 浏览器下正 常显示和运行，没有出现兼 容性问题 | 无 |
| TC02 | 用 Firefox 浏览器访 问网站 | 通过 | 系统在 Firefox 浏览器下正 常显示和运行，没有出现兼 容性问题 | 无 |
| TC03 | 在 Windows 操作系 统下访问网站 | 通过 | 系统在 Windows 操作系统 下正常显示和运行，没有出 现兼容性问题 | 无 |
| TC04 | 在 Mac OS 操作系 统下访问网站 | 通过 | 系统在 Mac OS 操作系统下 正常显示和运行，没有出现 兼容性问题 | 无 |

**表** **4.4** **兼容性全测试结果**

以上缺陷已经按照严重程度和优先级进行了分类和排序，并已经分配给相应的开发人 员进行修复。在 V1.9 版本解决了90%的问题。

**第五章** 安装及使用

**1.** **环境要求**

|  |  |
| --- | --- |
| **包名** | **版本号** |
| axios | ^1.6.8 |
| core-js | ^3.8.3 |
| element-plus | ^2.7.2 |
| less | ^3.9.0 |
| less-loader | ^12.2.0 |
| vue | ^3.2.13 |
| vue-router | ^4.0.3 |
| vuex | ^4.0.0 |
| Node.js | - |

**表** **5.1** **环境要求**

**2.** **部署流程**

首先安装 Node.js 环境：npm install 接着构建项目：npm run build

键入如下指令即可运行：npm run serve

后端可直接通过<http://8.141.12.156:12138/api>访问，在百度网盘提供了详细的 API调用文档。

如果想自行部署后端，Go 服务编译成二进制文件后通过 Docker 部署运行，MySQL 和 Redis 使用主从模式部署。

**3. 典型使用流程**：

在院校库界面，用户可以看到全国院校列表，利用上方的模糊搜索框和组合筛选框，通过院校位置和学校层次筛选出心仪院校。点击查看专业，可以查看该校开设的所有专业，包括学院名称、专业名称、专业代码和培养方式，并支持搜索具体专业。进一步点击查看某专业，可查看最近一年的录取名单、分数分析、近几年招生人数、考试范围和考生分析。专业库主页类似院校库，可通过学科门类和类别筛选专业，点击开设院校按钮查看该专业的所有院校。输入目标专业、分数、省份和学校层次并调整优先级，点击 AI 择校，即生成可选择的院校列表。用户还可在专业库中选择专业，查看开设该专业的所有院校，并通过省份和学校层次筛选，选中两个学校后可横向对比分数线和招生人数。

**第六章** 项目总结

**1.** **项目协调与任务分解**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 丁锐锐 | 胡嘉辉 | 李逸冉 | 贾径懿 | 韩韬 |
| 组织协调 | 55% | 30% | 5% | 5% | 5% |
| 前端开发 | 40% |  |  | 60% |  |
| 后端开发 |  | 70% | 30% |  |  |
| 择校算法设计 |  |  | 70% |  | 30% |
| 模型训练 |  |  | 70% |  | 30% |
| 界面设计 | 50% |  |  | 50% |  |
| 数据库设计 |  | 50% |  |  | 50% |
| 数据筛选及清洗 |  |  |  |  | 100% |

**表** **6.1** **任务分解与项目协调**

**2.** **项目总结**

本项目通过深入的需求分析，明确了考生在择校过程中的痛点，提供了一个集信 息整合、筛选和智能推荐于一体的综合平台。在设计和开发过程中，充分考虑了系统 的性能和可用性，通过多层架构设计和高效的技术选型，实现了功能全面且用户体验 良好的考研择校辅助工具。未来，我们将继续优化和完善系统，提升信息准确性和推 荐效果，为广大考研学子提供更优质的服务。

**3.** **后续改进思路**

当前项目在数据量、数据时效性、页面设计以及前后端通信安全性方面存在一些不足。为提升系统的功能性和用户体验，后续考虑进行以下改进：

**3.1** **增加数据量与数据来源：**

数据扩充：与更多高校、教育机构合作，获取最新的招生信息、考试科目、录取分数线等数据，建立更加全面的数据库。

用户生成数据：鼓励用户提交和分享个人的考研经历、分数及录取情况，丰富数据来源并增强数据的实用性和可靠性。

**3.2** **提升数据时效性**：

定期更新机制：建立自动化数据更新系统，定期从权威来源同步最新数据，减少信息滞后。

数据校验与清洗：引入数据校验和清洗机制，确保导入数据的准确性和一致性，提升数据的时效性和可靠性。

**3.3** **优化页面设计**：

响应式设计：优化网站的响应式设计，确保在不同设备和浏览器上都有良好的显示效 果和操作体验。

功能扩展：增加数据可视化功能，如使用图表展示录取分数趋势、招生人数变化等， 帮助用户更直观地了解信息。

**3.4** **增强前后端通信安全**：

HTTPS 加密：升级前后端通信协议至 HTTPS，确保数据在传输过程中的加密性和安全性。

认证与授权：实施 OAuth2.0 等认证和授权机制，确保用户数据的安全访问和操作权限管理。

数据加密与防护：对敏感数据进行加密存储，使用防火墙、防 DDOS 攻击等安全措施，提升系统整体的安全防护能力。

**3.5** **技术架构优化**：

微服务架构：将系统逐步转向微服务架构，提高系统的可扩展性和维护性。

云服务与分布式架构：利用云服务和分布式数据库技术，提升系统的扩展能力和数据处理效率。

**第七章** 参考文献

[1] 王鹏. MySQL 数据库优化技术与实践[M]. 北京: 电子工业出版社, 2018.

[2] 潘月梅. 基于机器学习的智能高考志愿推荐系统[D]. 南京邮电大学, 2019

[3] 任建涛. 推荐算法在高考志愿填报中的应用研究[D]. 云南财经大学, 2019

[4] 徐硕. 基于深度学习的推荐算法研究[D]. 中国矿业大学, 2020.