基于X86的小型人机交互操作系统的研究与实践

学 院：信息科学与工程学院

专业班级：电子信息工程（东软大数据）1901班

姓 名：吕恒熙

学 号：20191118015

指导教师：

2023年5月

基于X86的小型人机交互操作系统的研究与实践

**Research and practice of X86-based small human-computer interaction operating system**

## 摘 要

本文介绍了一个基于X86架构的小型人机交互操作系统的设计与实现。该系统使用汇编语言和C语言结合编写，具有较高的自由度和可扩展性。在设计过程中，我们着重考虑了以下几个方面：

1. 用户体验和交互性：为方便用户使用，我们设计了直观的图形用户界面（GUI）和简洁的操作方式。可通过鼠标、键盘等设备进行操作。操作系统支持常见的文件系统和命令行。在设计过程中，在保证系统稳定运行的同时尽可能提高系统响应速度，让用户可以体验到良好的操作感受。
2. 系统性能的优化：由于小型操作系统的硬件资源有限，我们优化了系统内部的代码和算法，节省资源并提高了效率。系统采用先进的页面管理算法，能够保证系统能够平稳运行，稳定性好，响应速度快。
3. 可扩展性：我们设计的小型操作系统支持多种设备和平台，可以轻松扩展到各种嵌入式和物联网应用场景。同时，我们考虑到接口的标准性和兼容性，让系统能够适应不同的硬件环境和软件的协议。

该系统在设计之初，就针对嵌入式物联网设备的需求，力求实现更高的效率和功率优化。除此之外，为了确保系统的稳定性和安全性，我们进行了一系列的测试和评估，验证了系统的可行性和有效性。最终的结果表明，该系统能够满足小型嵌入式设备的需求。

关键词：X86，操作系统，人机交互，可扩展性，用户体验。

**Abstract**

This paper introduces the design and implementation of a small human-computer interaction operating system based on X86 architecture. The system is developed using a combination of assembly language and C language, providing high degree of flexibility and scalability. In the design process, we focus on the following aspects:

1. User Experience and Interaction: To facilitate user usability, we have developed an intuitive graphical user interface (GUI) and simple operation methods. The system supports common file systems and command lines, and can be operated via devices such as mouse and keyboard. In the design process, we ensure that the system runs stably and efficiently, and optimize the response speed to provide users with a good operating experience.
2. System Performance Optimization: Due to the limited hardware resources of small operating systems, we optimize the internal code and algorithm of the system to save resources and increase efficiency. The system uses advanced page management algorithms to ensure smooth operation, stability and fast response speed.
3. Scalability: The small operating system we designed supports a variety of devices and platforms, making it easy to deploy in various embedded and IoT application scenarios. At the same time, we consider the interface standardization and compatibility, making the system adaptable to different hardware environments and software protocols.

This system is designed for embedded IoT devices, striving to achieve higher efficiency and power optimization. In addition, we conducted a series of testing and evaluation to validate the feasibility and effectiveness of the system, ensuring its stability and security. The results show that this system can meet the needs of small embedded devices, and has certain commercial prospects.

**Key words:** X86, operating system, human-computer interaction, scalability, user experience.

目 录

[摘 要 I](#_Toc103662217)

[Abstract II](#_Toc103662218)

[绪 论 1](#_Toc103662219)

[1 系统分析 2](#_Toc103662220)

[1.1 需求分析 2](#_Toc103662221)

[1.2 可行性分析 2](#_Toc103662222)

[1.2.1 技术可行性 2](#_Toc103662223)

[1.2.2 操作可行性 2](#_Toc103662224)

[1.3 相关技术 2](#_Toc103662225)

[1.3.1 B/S 结构 2](#_Toc103662226)

[1.3.2 VUE框架 3](#_Toc103662227)

[1.3.3 Apache ECharts 3](#_Toc103662228)

[1.3.4 MySQL数据库 3](#_Toc103662229)

[2 系统设计 4](#_Toc103662230)

[2.1 系统结构设计 4](#_Toc103662231)

[2.2 数据库设计 4](#_Toc103662232)

[2.2.1 数据库实体图 5](#_Toc103662233)

[2.2.2 数据库表设计 5](#_Toc103662234)

[3 系统实现 7](#_Toc103662235)

[3.1 信息显示模块实现 7](#_Toc103662236)

[3.1.1 登录模块实现 7](#_Toc103662237)

[3.1.2 图表显示模块 7](#_Toc103662238)

[3.1.3 权限管理模块 8](#_Toc103662239)

[3.1.4 日志显示模块 8](#_Toc103662240)

[3.2 数据存储模块实现 9](#_Toc103662241)

[3.3 数据处理模块实现 11](#_Toc103662242)

[3.3.1 数据清洗 11](#_Toc103662243)

[3.3.2 数据处理 11](#_Toc103662244)

[3.4 后台管理模块实现 12](#_Toc103662245)

[4 系统测试 13](#_Toc103662246)

[4.1 系统测试的目的 13](#_Toc103662247)

[4.2 系统测试用例 13](#_Toc103662248)

[4.2.1 用户功能测试 13](#_Toc103662249)

[4.2.2 测试结果分析 16](#_Toc103662250)

[5 研究或成果与社会、经济的关系 18](#_Toc103662251)

[结 论 19](#_Toc103662252)

[致 谢 20](#_Toc103662253)

[参考文献 21](#_Toc103662254)

## 绪 论

随着计算机技术的不断发展，操作系统作为计算机系统的核心部分，扮演着越来越重要的角色。不仅为用户提供了高效、稳定的计算环境，同时也促进了许多科学技术的发展和创新。为了满足不同用户的需求，现代操作系统的设计趋向于可定制性和可扩展性，允许用户进行自定义配置和扩展编程。在当前的多任务并行处理和虚拟化技术下，一款小型、高效的操作系统对于提高计算机性能和应用效率至关重要。

本文介绍了一款基于X86架构的小型操作系统，旨在提供稳定性、高效性、灵活性和可定制性等多方面的优点。该系统融合了汇编语言和C语言的编程思想，并采用多种算法和图形学原理，支持多种输入输出设备和自定义配置界面。该系统还具有多任务并行和虚拟机技术，可以在桌面和嵌入式系统等多种环境下运行，具有广泛的应用前景。

本文采用了实验方法对该操作系统的性能进行了测试和评估，并将结果进行了分析和比较。实验结果表明，该操作系统具有明显的优点，如高效、稳定、易于使用和可扩展性，可以在多串场合下得到应用。本文可以为操作系统的设计与应用提供有益的参考。

本文的主要结构如下：第一章介绍了操作系统的概念、发展历程和设计趋势；第二章详细描述了本文提出的操作系统的系统结构和功能设计；第三章介绍了操作系统的实现；第四章介绍了测试方法和功能测试；最后一章总结了本文所做的工作，并对未来的研究提出了一些建议和展望。

## 系统分析

* 1. 需求分析

IPL操作系统启动模块

计算机上电之后，主板厂家提供的BIOS进行开机自检，然后将顺位第一的储存设备引导扇区加载到内存地址为0x7c00处进行加载，而顺位第一的储存设备的主引导扇区就是IPL，IPL的主要作用就是加载操作系统，将操作系统程序从储存设备读入到内存当中，完成启动操作系统。

分段式内存管理模块

分页主要用于实现虚拟内存，从而获得更大的内存空间，而分段是为了使程序和数据可以划分为逻辑上独立的地址空间并且有助于共享和保护，基于X86的小型操作系统实现采用分段式内存管理，更有利于后续的开发过程。

中断处理使用外设

基于X86的小型人机交互系统使用了图形界面，鼠标的移动是需要中断的处理来实现的，所以，必须设计中断，引入GDT（全局段号记录表）和IDT（中断记录表），利用结构体来实现读写GDT和IDT。

多任务

现代操作系统支持操作系统必不可少，定义任务状态段，任务切换功能。

1. 图像界面

基于X86的小型人机交互系统原生支持图形界面，采用c语言和bios提供的画面绘制API来实现。

* 1. 可行性分析
     1. 技术可行性

地震大数据分析系统所用到主要开发语言为VUE和Python。我所使用的IDE主要是HBuilderX以及Visual Studio Code分别来开发VUE和Python，Python不但作为整个系统的后端，同时也可以方便的插入数据分析相关的内容，是最适合进行数据处理的语言。VUE是一门前端语言，在大四期间我曾负责过一款游戏的网页开发，所以本次项目使用同样的前端语言开发，大大降低了开发前期准备工作，有利于将更多的时间投入到系统的内容设计以及前端的人机交互设计中去。

* + 1. 操作可行性

纵观整个数据流程，从最初在各个地震台公布数据开始，全球的地震数据均是开源的，而我国的地震数据中心有关于地震数据的公布相关规定，地震数据是被允许公开，且随意进行以科研为目的的项目研究的，由于本次项目并未涉及到经济价值产出，所以原则上数据流通是合理合法的[3]。在网页端部分，整个系统需要对于传输过来的数据进行简单的数据筛选，随后将筛选内容发送到前端的各个部件处将其可视化，这一过程不但符合人脑逻辑，在开发上也降低了代码攥写的难度，操作性相较于同类其他的数据分析系统较为简单[4]。

* 1. 相关技术
     1. B/S 结构

B/S结构全程为 Brower/Server结构（浏览器/服务器）结构。该结构为一种系统设计结构，意在梳理服务器和浏览器之间的数据通路。基于这种结构设计的系统前端代码与后端代码分离，减少每个部分的开发周期。这样的结构设计使得用户所处在的浏览器端负载压力变得轻量，可以减少对于用户端资源的占用。毫无疑问设计也能让系统维护变得更加轻松。

* + 1. VUE框架

VUE框架全名为VUE.js。顾名思义，这是一个基于JS语言的前端框架。它能用于构建用户界面并具有渐进式的特性。与一些老旧的大型web框架不同，VUE从设计之初就时可以自底向上逐层应用的。VUE 的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持类库结合使用时，VUE 也完全能够为复杂的单页应用提供驱动。这也是地震大数据分析系统最终选择VUE作为配合服务器端的前端框架的重要原因。

* + 1. Apache ECharts

Apache ECharts是一个基于 JavaScript 的开源可视化图表库。作为一个由百度开发，适用于网站开发的JS库，它具有许多我们所需要的图库特性。从数据管理的角度上，使用ECharts可以自动化的对数据进行过滤、回归、聚类等操作。ECharts的渲染引擎和丰富的图表类型是我们使用它和VUE进行结合的主要原因。

* + 1. MySQL数据库

MySQL作为最经典的关系型数据库，在强关联数据的处理上有着得天独厚的优势。免费、跨平台、极好的兼容性也是很多产品选择它作为其数据库的理由。考虑到MySQL的增删改查相对直白，也符合地震数据的特性，故该系统放弃使用MongoDB等数据库类型，采用MySQL数据库作为系统整体数据存储的数据库语言。

## 系统设计

* 1. 系统结构设计

本系统的结构设计以B/S结构为模型，主要分为用户端和服务器端。服务器端运行网站代码，进行数据处理等各相工作，并与前端进行实时交互渲染。而用户端分为两大部分：客户端和管理员端。客户端的内容主要是调取服务器资源，整理出所需要的地震数据，将其显示出来，而管理员端则可以使用VUE的特性来实时修改与维护整个系统的各类信息和功能，避免信息修改的时候产生应用空窗期。

地震大数据分析系统的功能结构图，如图2-1所示。

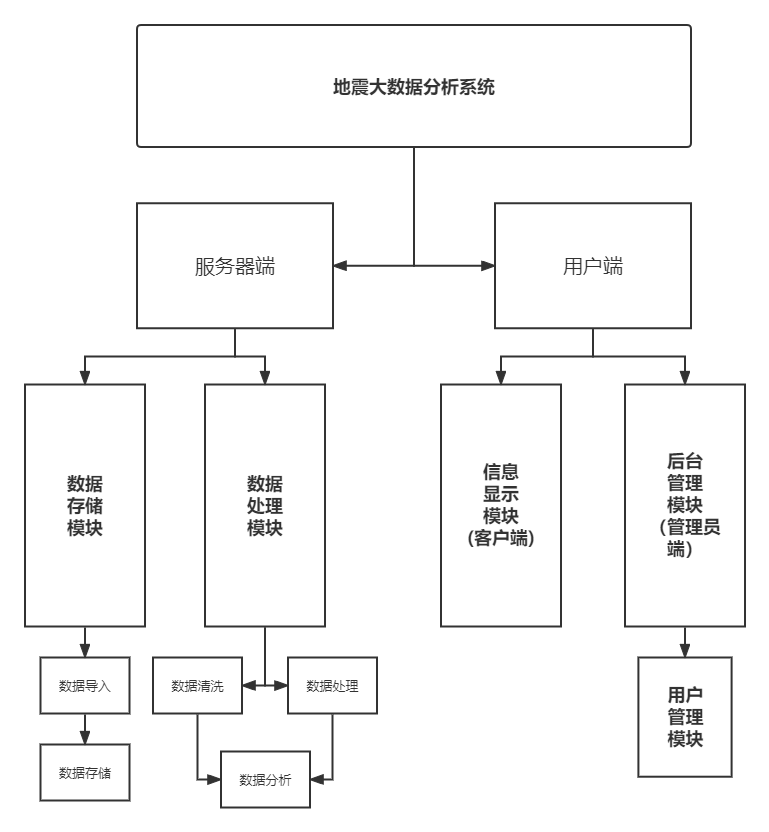


图2-1 系统功能结构图

* 1. 数据库设计

在地震大数据分析系统的后端内部，我计划使用了MySQL作为关系型数据库后台软件，由于本系统可调用的地震数据集本身已具有完整性和强关联性，所以只需在数据库设计中根据数据类型进行设计即可。

* + 1. 数据库实体图

数据库E-R图能够很好的反应数据库内设计实体的有效结构，从视觉上高效合理的对数据库内各项数据进行分析解读。在地震大数据分析系统中由于不同于标准的JAVAWEB项目需要创建实体类，在该系统里主要利用数据库调用的数据是地震数据集以及用户数据。地震数据集中的数据实体主要由多个测量数据构成，而用户数据则主要涵盖地震数据系统的管理员权限信息，数据平台用户权限信息。

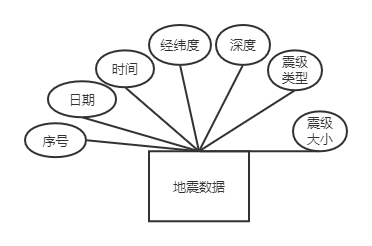
以下是地震数据实体图，如图2-2所示。

图2-2 地震数据实体图

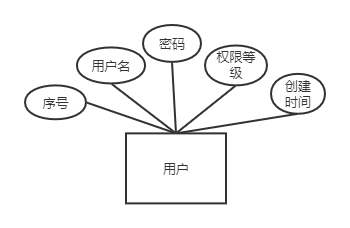
在用户管理模块中，设置一个用户信息实体图，如图2-3所示。

图2-3 用户数据实体图

* + 1. 数据库表设计

通过用户需求分析、项目需求管理以及对地震数据分析系统的综合判断，在实体图的基础上我们对数据库内数据的映射关系进行表格展示，表格展示不但能将各类数据表现更加显著，同时让地震大数据更简易的被进行识别和处理。

用户数据信息表（user表）详细设计如表2-1所示，包含序号、用户名、密码、权限等级以及创建时间。

表2-1 user表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 长度 | 约束 |
| id | int | 32 | NOT NULL |
| name | varchar | 32 | DEFAULT NULL |
| password | varchar | 32 | DEFAULT NULL |
| permissions | int | 2 | DEFAULT NULL |
| createDate | datetime | 0 | DEFAULT NULL |

特别注意的是，由于createDate列的数据结构为datetime，所以不存在长度的概念，只会以时间形式被记录下来。

地震数据表（data表）详细设计如表2-2所示。表中包含序号、日期、时间、经度、纬度、深度、震级类型以及震级大小。

表2-2 data表

|  |  |  |  |
| --- | --- | --- | --- |
| 列名 | 数据类型 | 长度 | 约束 |
| id | int | 32 | NOT NULL |
| Date | varchar | 32 | DEFAULT NULL |
| Time | varchar | 32 | DEFAULT NULL |
| longitude | int | 2 | DEFAULT NULL |
| latitude | int | 2 | DEFAULT NULL |
| Depth | int | 2 | DEFAULT NULL |
| Type | varchar | 32 | DEFAULT NULL |
| Size | int | 2 | DEFAULT NULL |

## 系统实现

* 1. 信息显示模块实现
     1. 登录模块实现

访问相对应网站的8001端口，整个地震大数据分析系统的门面网站即为如图3-1所示的登录界面，整个界面干净整洁，当用户输入正确的用户名密码后，即可登录进网站中。

图3-1 登录界面图

* + 1. 图表显示模块

ECharts是一个集成化的多图表显示工具集。它不但可以根据使用者需求来灵活调动其访问性，同时还可以动态的对生成的图表进行视觉设计上的改动，这不但让开发者的工具量大大降低，同时也让系统用户不再只是追求于一个简单的数据收集工作，还可以对交互、性能、数据处理等方面进行更深层次的探究。

如图3-2所示，在页面里会显示我们后台数据中的地震数据，这种数据大屏的形式，大大提升了数据分析的效率。

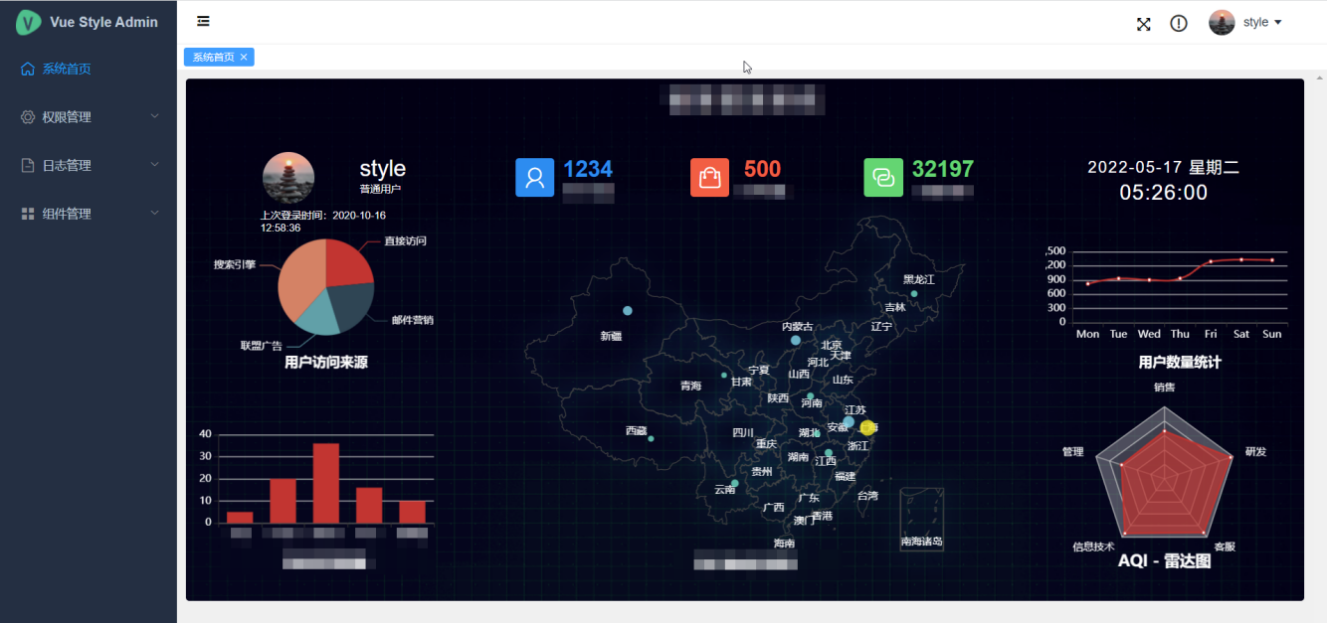


图3-2 ECharts大屏图

* + 1. 权限管理模块

地震数据分析系统不但有普通用户可以访问，管理员同样也能进行访问并对网站内容进行管理，这里的角色管理可以修改角色是否能正常进入整个系统进行查看，而菜单管理则如图3-3所示，对需要管理的人员数据进行了权限控制，使其在登录网站界面的时候不会看到权限管理模块，当然更无权力对用户权限进行修改。

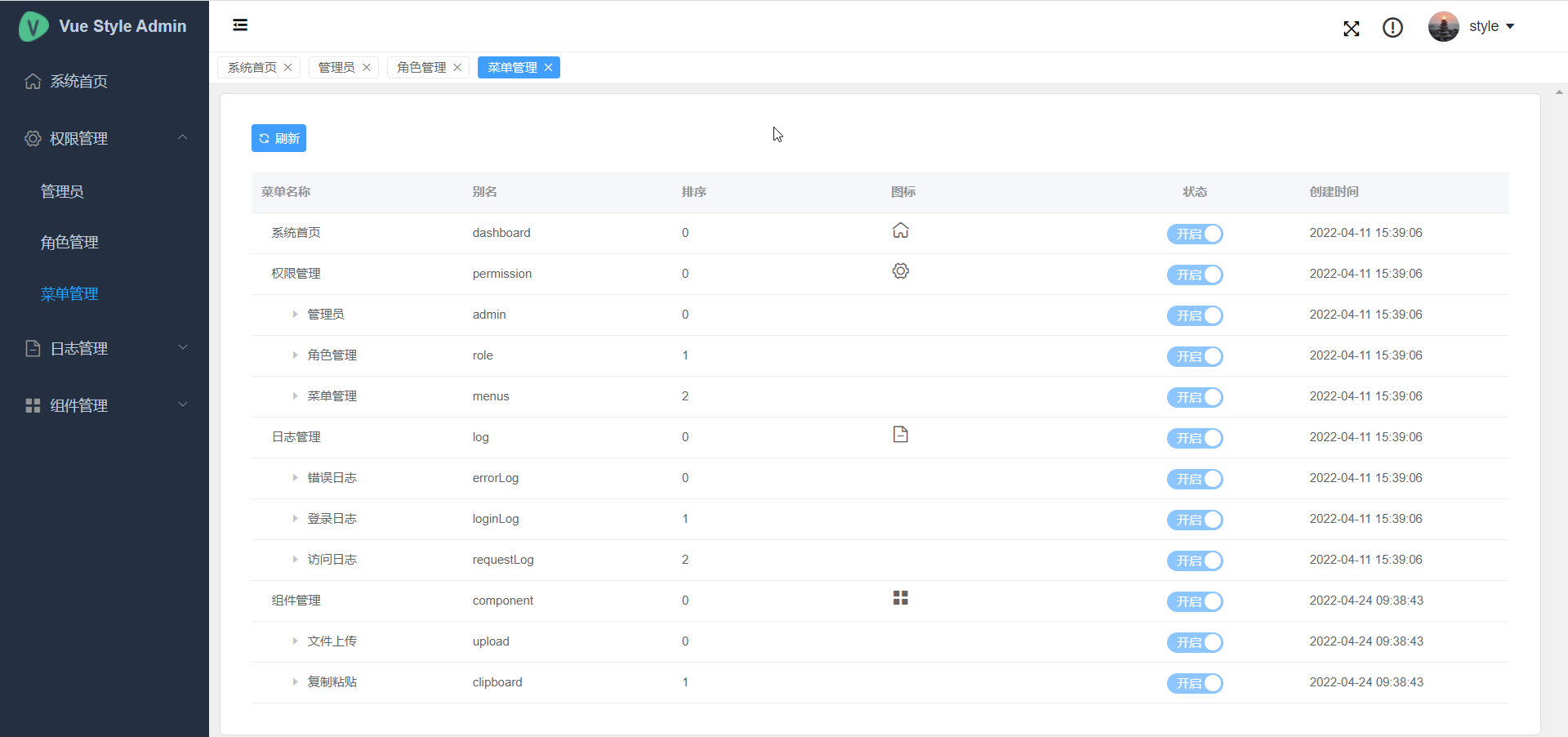


图3-3 权限管理图

* + 1. 日志显示模块

考虑到整个系统有多层次的人员管理、多个数据信息库，所以增加了一个日志显示模块，在后台设置中记录系统的错误日志、系统登录日志以及访问日志，如图3-4所示，错误日志的显示可以提高管理员对于网站的维护效率，降低维护成本。

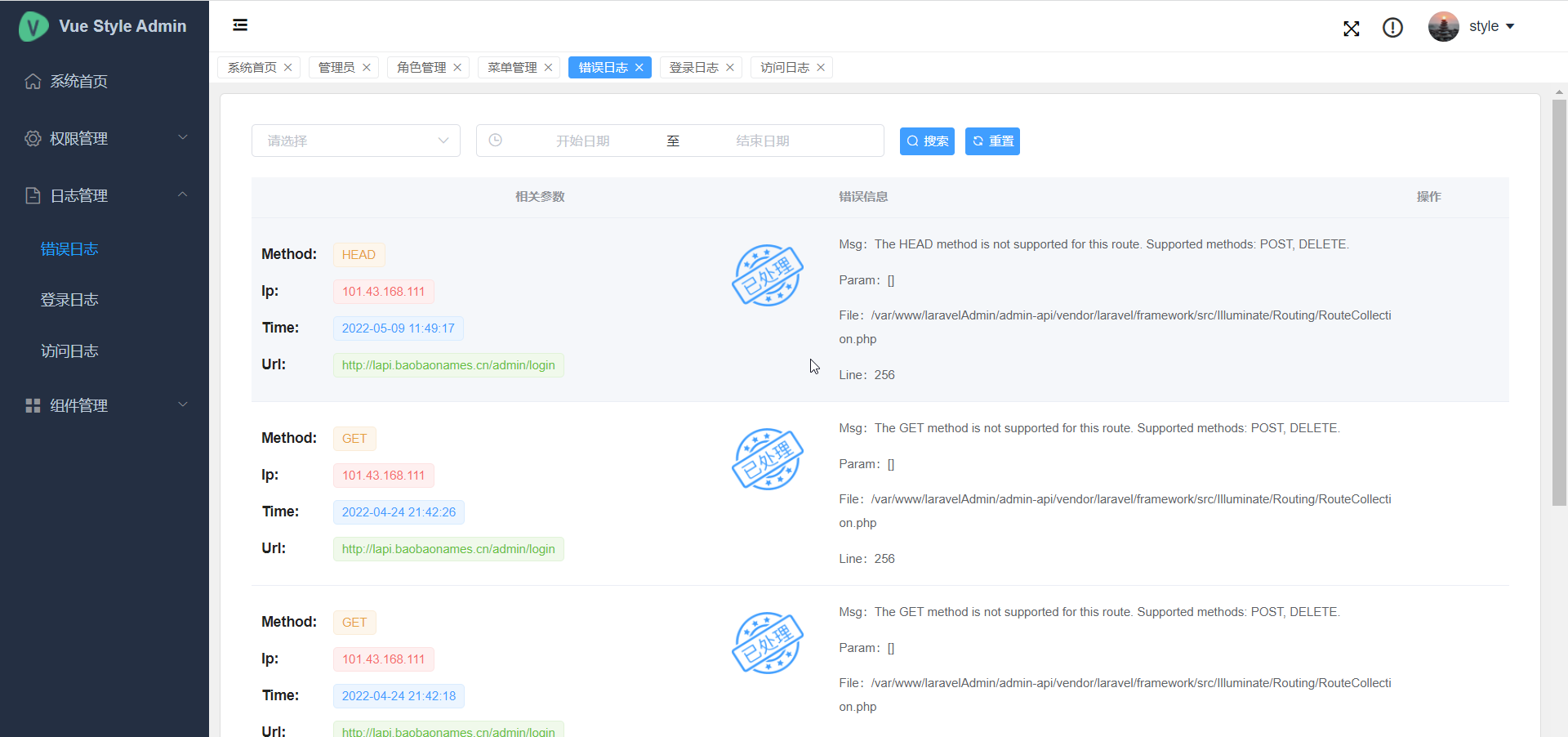


图3-4 权限管理图

* 1. 数据存储模块实现
     1. 数据导入

对于当前的数据导入相对简单，我们人工对于地震大数据集进行了筛选，最后暂时使用国家地震科学数据中心提供的中国大陆及邻区震源机制数据集（2009-2021年），如图3-5所示，之后将采用python脚本的方式对实时地震大数据在中国地震台网上进行收集，保持每日更新。



图3-5 中国大陆及邻区震源机制数据集（2009-2021年）

* + 1. 数据存储

对于流程化的地震数据分析平台，目前我们使用的是MySQL进行数据的存储。但是由于数据集是以excel文件形式被下载到本地或服务器，所以不能直接转存至数据库中，这里我们借用python中的pandas库，对xls文件内容进行读取，通过pandas对excel内文件进行定制化的格式修改，最后变成一列列的数据流，并将其转存至MySQL数据库中。具体核心代码如下：

import pandas as pd

def excel\_one\_line\_to\_list():

df = pd.read\_excel("402序列.xlsx", usecols=[1, 2],

names=None) # 读取项目名称列,不要列名

df\_li = df.values.tolist()

# print(df\_li)

# update team set room = xxx where teamid = yyyy;

ans\_lst = []

for i in df\_li:

sql = "update team set room=\'{1}\' where teamid={0};".format(i[0][4:], i[1])

ans\_lst.append(sql)

print(ans\_lst)

f = open("test.sql", "w")

for i in ans\_lst:

f.write(i + '\n')

df = pd.read\_excel("Earthquake.xlsx", usecols=[1, 3],

names=None) # 读取项目名称列,不要列名

df\_li = df.values.tolist()

ans\_lst = []

for i in df\_li:

sql = "update team set room = \'{1}\' where teamid = {0};".format(i[0][4:], i[1])

ans\_lst.append(sql)

for i in ans\_lst:

f.write(i + '\n')

f.close()

if \_\_name\_\_ == '\_\_main\_\_':

excel\_one\_line\_to\_list()

* 1. 数据处理模块实现
     1. 数据清洗

对于我们所使用的震源机制数据集，其中的数据相对来说仍旧过于详细，不但包含我们常用的几项数据类型，同样还包含了矩心深度、节面走向、倾角、滑动角等诸多详细数据。由于本身地震大数据分析系统的数据量将会越来越庞大，所以我们在数据从数据库发送到前端之前，便对整个数据库里的信息进行数据筛选，最后得出的结果如图3-6所示，只剩下我们需要显示在前台的信息。

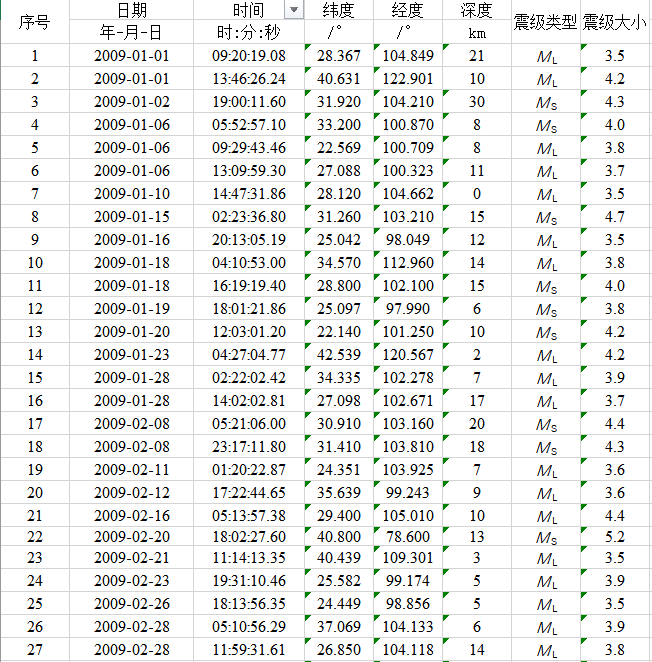


图3-6 数据结果筛查图

* + 1. 数据处理

地震数据分析系统为使用DOM来进行动态显示的系统，整个系统的数据嵌入均有后台发送相关JSON数据来实现，而前台的ECharts图表仅需将dataset（数据集）格式设置好，对后台发来的数据一一对应显示即可。我们使用python脚本读取SQL相关数据，将其制作成JSON数据，便于前端图表进行调用，具体核心代码如下：

if \_\_name\_\_ == '\_\_main\_\_':

# 开始操作

# 将原来地震的数据全部按照某种形式统计出来

data\_dict = main()

Earthquake\_list = []

# 转化格式

# 将数据做成列表的样子 {name：xxx,value：xxx}

for key, values in data\_dict.items():

vars\_dict = {}

vars\_dict['name'] = key

vars\_dict['value'] = values

Earthquake\_list.append(vars\_dict)

with open('Type.json', 'w') as f:

# 输出

json.dump(Earthquake\_list, f)

* 1. 后台管理模块实现
     1. 接口模块

B/S模式需要将服务器端和客户端做一个前后端分离，在地震数据系统的完整开发周期最开始便定下了接口规格，整个系统中的所有数据接口，都使用SWAGGER-API文档进行API接口管理，统一JSON格式后对于将来系统迭代，地震数据更新时都会极大程度减少工作量，如图3-7展示了部分API通用接口，接口模块的设计更是为了可以多人协同开发地震数据系统着想的。

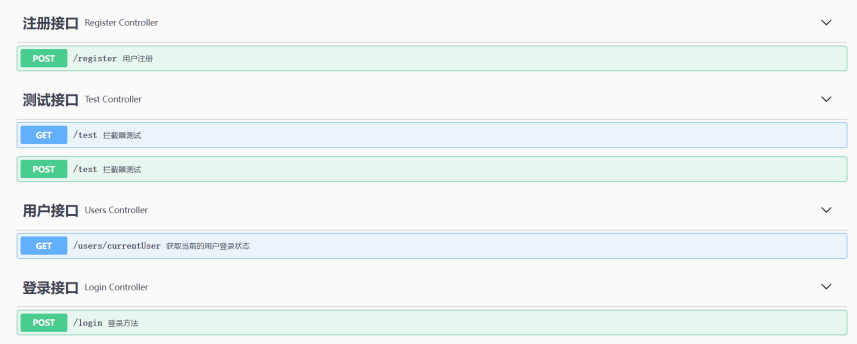


图3-7 API接口文档图

## 系统测试

* 1. 系统测试的目的

系统测试是检验整个系统开发工作的最后工序。对于每一项成熟的系统，都离不开系统测试这一步骤。系统测试能通过设置完善全面的测试项，将开发遗留下来的漏洞和BUG进行查漏补缺。并在上线前做最后的安全检查。

本次系统测试要对整个地震大数据分析系统进行详细的测试。主要针对的问题有，地震数据分析系统的前端中是否有不可逆的交互问题，是否存在交互逻辑十分明显的漏洞，项目部署在服务器上是否在配置上与本地开发环境有区别，是否会产生影响，用户登录注册是否有逻辑漏洞，图表的显示是否严格遵循开发人员给予其的规则，数据库调用是否正常。

在进行过系统测试后，能基本确保系统可以平稳运行，在较长时间内不会出现根本性问题，在一定程度上解决了项目开发不便于Debug的问题，这也能大大提升地震数据分析系统的使用体验。

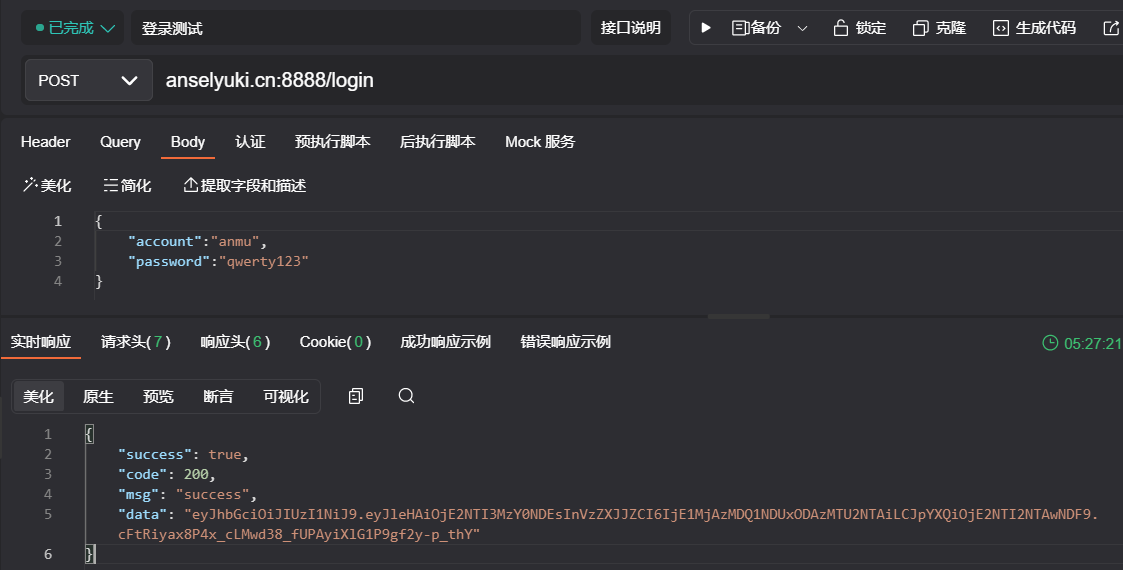
* 1. 系统测试用例
     1. 用户功能测试

用户登录失败测试用例，测试用例如下表4-1所示。

表4-1 用户登录测试

|  |  |
| --- | --- |
| **用例编号** | ST0001 |
| **测试项目** | 用户登录功能 |
| **测试标题** | 用户在用户名、密码错误时能否登录系统 |
| **操作步骤** | 打开浏览器，输入相应地址进入系统登录页面输入用户名：123456  输入密码：123qweasd  点击登录 |
| **预期结果** | 登录失败，用户无法进入系统 |
| **实际结果** | 登录失败，用户无法进入系统 |

在主界面中，点击主页面的右上角的登录按钮即可进行登录操作，当使用者通过前端输入的数据与后端服务器数据进行核对后，数据匹配，后端服务器则会向前端返回一个json文件，记载着当然用户成功登录网站，并给予其一个唯一的token值。如图4-1所示。我们对登录接口进行测试，可以看到我们输入了正确的用户信息后，产生了data，将这个data数据添加进前端记录，附加在之后所有操作的Header中，则可以确保用户的操作权限是不会越界的。

图4-1 登录测试图

在进行了用户鉴权后，所有的信息显示将不再受到限制，这可以确保地震大数据系统内的所有数据均可正常访问。

用户登录成功后测试图表显示功能用例，测试用例如下表4-2所示。

表4-2 用户显示测试

|  |  |
| --- | --- |
| **用例编号** | ST0002 |
| **测试项目** | 图表显示功能 |
| **测试标题** | 用户在用户名正确登录后能否能看见图表 |
| **操作步骤** | 打开浏览器  输入相应地址进入系统登录页面输入正确的账号密码  点击登录后观察是否显示了图表 |
| **预期结果** | 登录成功，主界面出现ECharts图表 |
| **实际结果** | 登录成功，主界面出现ECharts图表 |

用户登录成功后对图表数据进行导出测试，测试用例如下表4-3所示。

表4-3 数据导出测试

|  |  |
| --- | --- |
| **用例编号** | ST0003 |
| **测试项目** | 图表数据导出功能 |
| **测试标题** | 用户在图表中能否能看见导出按钮并顺利导出图表 |
| **操作步骤** | 打开浏览器  进入系统  查看相关页面  观察是否正确显示了图表  点击导出，图表中各项导出功能是否可用 |
| **预期结果** | 导出成功，生成PNG表格图片 |
| **实际结果** | 导出成功，生成PNG表格图片 |

管理员登录失败测试，测试用例如下表4-4所示。

表4-4 管理员登录测试

|  |  |
| --- | --- |
| **用例编号** | ST0004 |
| **测试项目** | 管理员登录功能 |
| **测试标题** | 管理员在输入错误的用户名和密码时能否登录系统 |
| **操作步骤** | 打开浏览器  进入系统后台  在登录页面输入用户名：admin123，输入密码：12345  输入验证码后，点击登录  查看登录结果 |
| **预期结果** | 登录失败，管理员无法进入系统 |
| **实际结果** | 登录失败，管理员无法进入系统 |

管理员登录成功后登录后台进行数据测试，测试用例如下表4-5所示。

表4-5 管理员更新数据测试

|  |  |
| --- | --- |
| **用例编号** | ST0005 |
| **测试项目** | 数据更新功能 |
| **测试标题** | 管理员在登录成功后更新数据 |
| **操作步骤** | 打开浏览器，输入相应地址进入系统后台  在登录页面输入管理员用户名和密码  点击登录  点击更新按钮，对数据库数据进行更新 |
| **预期结果** | 系统提示成功，前台数据得到更新 |
| **实际结果** | 系统提示成功，前台数据得到更新 |

模拟数据传输过程中出错的测试用例，测试用例如下表4-6所示。

表4-6 异常测试

|  |  |
| --- | --- |
| **用例编号** | ST0006 |
| **测试项目** | 数据异常报错 |
| **测试标题** | 若项目数据异常则弹出报错信息 |
| **操作步骤** | 打开浏览器，输入相应地址进入系统后台  在登录页面输入用户名和密码  点击登录  对数据库数据进行错误修改，并进行更新 |
| **预期结果** | 系统提示-999，系统异常 |
| **实际结果** | 系统异常，并没进行提示 |

* + 1. 测试结果分析

在以上的测试用例中，有诸如管理员登录失败、用户导入数据等日常用户会进行的一系列操作，受益于在系统开发初期已经考虑的相对周全，在用户登录、系统查询、数据修改等诸多方面系统测试均顺利通过。操作步骤符合测试逻辑，结果基本满足预期效果，基本符合预计现象，没有出现较大意外。当然，我们仍要看到有诸多不足。

在测试用例的设置上，我对每一个项目中可能出现的问题进行的分析，最后得出了六项可能性分支，基本能保证六项测试结束后整个地震大数据分析系统能够平稳的运行下去，同时这也符合边界值测试的测试理念，是符合逻辑的测试项目[5]。

在6项系统测试中，最后一项测试是对系统发生异常时弹出提示的测试。在接口文档中我们将其设置为一旦产生报错信息则弹出-999的异常代码，原意是类似于通过垃圾回收或者异常机制的方式强行阻止系统继续运行下去，但最后测试结果中当我们对xls文件中的数据进行破坏，从python的格式转化开始就开始报错[6]，数据库数据也出现了导入异常，虽然因为不会覆盖原先的数据所以没有显示错误信息，但在数据返回的部分查看接口则已经变成-999了。这是一个小BUG，目前计划是对于数据传输的过程进行更严密的筛选工作，防止在数据导入的过程中产生错误代码，加入一些排查代码在原先的导入代码里，这样可以更加有效的规避掉数据层面出错的问题。

## 5 研究或成果与社会、经济的关系

从2009到2021年，全世界的板块移动并不“冷静”。不会有人忘记唐山大地震，汶川大地震给国家带来了多大的经济损失，有多少条无辜的生命因为地震这种天灾而丧失继续生存下去的权利。身处在亚欧板块上的我们应该很清楚的认识到，地震也好，板块移动也好，都是无时不刻在发生的事情，一些震感不强烈的地震似乎来的这么的平常，以至于让有些人对于地震这两次失去的敬畏。地震大数据分析系统，不单单只是一个分析系统这么简单，我将2009年至2021年的所有震级高于3.5级的地震信息数据集收集并展示出来，也是希望能告诉每一个人，地震离我们真的不远，也并不少见，在一次次对于数据的筛查中我们不难发现，在评估大型项目的风险和安全时，诸如建房、搭建大型设施等与地基有强关联性的项目，你必须要对地震这件事重视起来，不单单是在地震带上的人们需要关注，哪怕是远离地震高频区的其他人，也要无时无刻提高安全意识，尽可能的避免天灾的发生[7]。地震数据分析系统一部分适用于需要一些简单地震数据，且对可视化有一些需求的人群，也适合一切需要进行地震安全科普的机构组织个人。这一份数据拿出去，可以和防灾开设情况，地震救援物资的储备等信息进行交叉分析，这也能让防震防灾工作能更好更快的进行下去。

社会与经济的效益永远都是相辅相成的，当防灾减灾逐渐走入人心，让大家的预防地震意识得到极大的提高之后，地震大数据分析系统就可以更进一步的，对更伟大的目标奋进，那就是参与对地震预警、地震预报的研究工作中去。到目前为止即便是国际上公认的Kiwi算法，也不能有一众压倒性的说服力说法其他人相信现在的技术可以对地震进行10秒到3分钟内的预警[8]。在地震预测这一点上我也深信不疑。我相信整个系统在不断的对技术和功能进行更迭的同时，也能为现在的地震学的进步出一份力所能及的力量。

## 结 论

历经前前后后约莫两个多月的时间，对于整个地震大数据分析系统来说，我完成了从最早的需求分析，对各项技术栈探索的早期阶段，随后跟指导老师逐渐沟通交流，基本完善了整个项目，直至最后将整个系统顺利的进行产出，并加以测试来对其进行改进。整个项目的开发过程是充实且收获颇丰的。

从最早确认要做数据分析系统的时候开始，我便开始思考需要使用何种类型的编程语言来进行开发，又是用哪种框架能比较好的承载这个系统的运营维护[9]。最后我发现了Python作为一种脚本语言，不但对于网站搭建有着极强的兼容性和功能性，适合对各类科学数据进行处理，还可以用来作为网站框架，有很多成熟的开源框架可以使用，基于这种想法，地震大数据分析系统的项目就基本确定是以Python为主的项目了[10]。而在前端设计方面，我考虑到VUE的渐进式框架是非常好的可以动态规化显示效果、实时进行操作、易于修改的这么一种前端框架，所以在进行深度的了解之后，整个地震大数据分析系统就成为了Python+VUE这么一种结构的项目。在这其中我发觉了Python对于excel文件的处理，JSON文件的处理包括对数据库SQL语句都有较强的兼容性，也得益于它的兼容性，本次项目的数据流转变得异常顺利[11]。

从最早开始关注中国地震台网，希望能从中国地震台网上爬取部分信息，到后面发现国家地震数据中心的测震系统数据集[12]，我在其中学到了很多关于数据获取数据导入的知识，也同时明白了任何一项大数据的诞生都不是轻轻松松的，数据生产者往往需要进行长期大量重复的工作才能最后得到一份信噪比高的、可靠性高的、相互之间误差较小的数据。这份数据非常的来之不易[13]。

在数据处理阶段，我自己使用了Python作为数据处理的工具，也同时发现了ECharts这么一个快捷的JS图文库。利用这个库封装好的各种功能，可以高效快速的实现各种曾经觉得非常复杂的事情，比如在前端界面上实现筛选表的整理工作、可以更改黑暗模式、直接显示dataview等等功能。这些在我认识到ECharts.js的好用之前认为是靠自己一个人很难实现的功能，现在有了ECharts，整个项目也得以锦上添花[14]。

## 致 谢

项目圆满结束，首先要感谢的是在整个毕设过程中给予我莫大宽容和技术支持的指导老师卓老师，从开始选题、查找数据，到中间开发周期的协商、最后的论文的攥写，她在我身上花费了诸多心思，由衷的感谢这段时间以来您的付出，希望未来还有机会我们还可以再一起畅聊一些有意思的项目和学习内容。

大学四年，岁月如梭，似乎刚入校门口的时光就在昨日，每一个人都好像有所变化，但无论怎么变化，我都认为这四年我在武昌首义学院的收获颇丰的。从刚入学以来什么编程都不会，到后面对ACM爱好者协会感兴趣，被金兰、张硕老师提携，我在协会里学会了各种各种的竞赛知识，算法知识，代码知识，这短短的几年时间我基本已经掌握了很大一部分的编程技巧，培养起了我的计算机思维，这也让我能在毕业设计这一大学里最后的一项事务中划上一个对于大学四年完美的句号。

在此，还要感谢这几年来给予我帮助的各位同胞。其中有同为协会负责人的学长学姐，也有比我小的学弟学妹，这几年来我们一起共同进步，相互学习，让武昌首义的学习氛围更加浓郁，相信不管是我也好还是其他人都能在大学生活中享受学习、享受人生，并让这四年生涯作为自己未来的跳板，跃向更美好的未来。

最后特别感谢我的父母，感谢他们多年以来对我的照顾和关心。是你们无私的付出，不断培养我对计算机的浓厚兴趣，我才能有今天这般成绩。虽然这20多年里，我们有过吵闹，有过争执，但我们的目标都是为了我能有更好的未来，我能拥有更扎实的基础，事实证明你们是对的，是你们对学习的态度让我耳濡目染，让我能在书桌前耐住心思，不管是学习算法还是项目开发，一颗沉稳的心和坚持不懈的态度是能做成一切事情的基石。这都是你们功劳，再次表达我的感谢。

致谢已将结束，我一直希望我能给这个社会产生一些价值，对身边人带来一些帮助，这是我一直在坚持的信念，我希望自己能做到无限进步，这样才能在逆水行舟不进则退的世界里认清自己所处在的位置，不断完善自己的能力，让自己的能力能更好的造福社会，造福身边的每一个人，希望我的成果能给每个人都带去一些力所能及的价值，那么这将会是对我最大的认可。

## 参考文献

1. 赵辉.地震监测数据的Hadoop存储解决方案[J].华南地震.2020,40（3）：70-75.
2. 刘海,田智慧,王宗超等.基于Spark的地震数据处理与分析系统设计与实现[J].电脑编程技巧与维护.2022,(04):119-121+134.
3. 师天祺.基于Python的地震数据可视化[J].数字技术与应用.2022,40(03):180-182.
4. 张岩,李新月,王斌等.基于深度学习的鲁棒地震数据去噪[J].石油地球物理勘探. 2022,57(01):12-25+4.
5. 陈新房,刘义卿. 基于分布式系统的地震数据处理及可视化研究[J].科学技术创新.2021,(35):64-66.
6. 刘禹,李胜乐,李力.大数据平台的测震数据流实时可视化系统设计与实现[J].测绘地理信息.2022:1-8.
7. 唐刚.基于压缩感知和稀疏表示的地震数据重建与去噪[D].北京:清华大学，2010.
8. 王志浩.地震大数据机器学习平台[D].河北：防灾科技院,2021.
9. 陈建江.AVO三参数反演方法研究[D].山东：中国石油大学,2007.
10. 赵国峰,庞丽娜,李丽.国家地震科学数据中心的数据资源与共享服务[C].第三届地球物理信息前沿技术研讨会论文摘要集.2021:86-88.
11. 张俊威,孙友凯,刘芳等.地震资料处理流程大数据分析与应用[J].石化技术. 2021,28(06):136-138.
12. 崔兵.陆上地震数据采集节点设备现状与发展[J].中国石油和化工标准与质量. 2021,41(10):126-127.
13. 张培震,邓起东,张国民等.中国大陆的强震活动与活动地块[J].中国科学(D辑:地球科学).2003,(S1):12-20.
14. 常德宽,雍学善,王一惠等.基于深度卷积神经网络的地震数据断层识别方法[J].石油地球物理勘探.2021,56(01):1-8+230.
15. Li Juan, an ran, Li Yue, etc. Desert Noise Suppression for Seismic Data Based on Feature Enhancement Denoising Network[J]. [Izvestiya, Physics of the Solid Earth](https://scholar.cnki.net/journal/index/SSJD106935131361).Volume 57.Issue 6. 2021:935-949.
16. Oleg Elshin, Andrew A. Tronin. The Theoretical and Practical Foundations of Strong Earthquake Predictability[J]. Open Journal of Earthquake Research.2021,10: 17-29.