

# 论文检测全文报告

## 基本信息

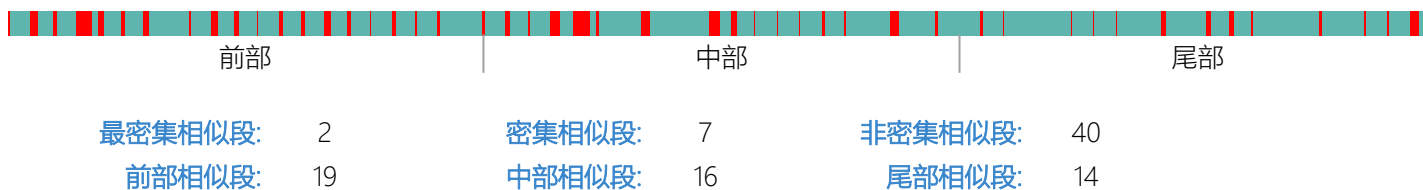
报告编号: 202205168290197222836F020

文档名称: A查重  
提交方式: 上传文档检测  
提交时间: 2022年05月16日  
正文字符数: 16946  
检测范围: 大雅全文库

## 总体结论

文献相似度: 14.56%  
文献原创度: 85.44%  
去除参考文献相似度: 14.56%  
单篇最大相似度: 0.65%  
去除本人已发表论文相似度: 14.56%  
单篇最大重复数: 112  
重复字符数: 2468  
最相似文献题名: 教育云环境下在线考试系统关键问题探讨与研究

## 相似片段分布



## 典型相似文献

### 相似图书

序号	题名	作者	出处	相似度
1	Web开发技术及其应用	王成良	北京: 清华大学出版社, 2007.12	0.64%
2	Vue.js 2.x实践指南	邹琼俊	北京: 清华大学出版社, 2020.03	0.54%
3	Vue.JS前端开发技术	王凤丽;豆连军	北京: 人民邮电出版社, 2019.03	0.54%
4	Vue.js前端开发实战	黑马程序员	北京: 人民邮电出版社, 2019.12	0.53%
5	Vue.js项目开发实战	张帆	北京: 机械工业出版社, 2018.08	0.5%
6	Web开发技术 第2版	王成良;祝伟华;柳玲	北京: 清华大学出版社, 2013.08	0.42%
7	JavaScript从入门到精通	明日科技	北京: 清华大学出版社, 2019	0.4%
8	Struts 2 Web开发学习实录	吴越人;樊双灵	北京: 清华大学出版社, 2011.07	0.35%
9	Flutter实战	弗兰克·扎米蒂;贡国栋;任强	北京: 清华大学出版社, 2020.07	0.35%
10	前端程序员面试秘籍	张容铭;爱创课堂	北京: 人民邮电出版社, 2018.12	0.34%
11	精通JS脚本之ExtJS框架	赵俊昌;祝红涛;吴越人	北京: 化学工业出版社, 2011.03	0.34%
12	Spring Boot开发实战 微课视频版	吴胜	北京: 清华大学出版社, 2019.09	0.34%
13	HTML5移动Web+Vue.js应用开发实战	吕鸣	北京: 清华大学出版社, 2020.05	0.34%
14	证券行业金融科技探索与实践 证券信息技术研究发展中心(上海)2018年课题报告精选	证券信息技术研究发展中心(上海)	格致出版社; 上海人民出版社, 2019.11	0.34%

15	大数据技术体系与开源生态	刘驰	北京：人民邮电出版社，2018.08	0.33%
16	电子政务与政府信息资源管理研究与探索	杨美沂	长春：吉林大学出版社，2018.05	0.33%
17	ASP.NET标准教程	常倬林	北京：化学工业出版社，2011.01	0.32%
18	电网防灾减灾应急管理系统建设与应用	林韩	北京：国防工业出版社，2009.05	0.31%
19	HTML5+CSS3+JavaScript+jQuery程序设计基础教程	闫俊伟;耿强	北京：人民邮电出版社，2018.08	0.29%
20	Flash二维动画设计与制作	刘进军;肖建芳;丰伟刚	北京：清华大学出版社，2011.05	0.25%
21	NoSQL数据库原理与应用	王爱国;许桂秋	北京：人民邮电出版社，2019.03	0.23%
22	Vue.js前端开发 快速入门与专业应用	陈陆扬	北京：人民邮电出版社，2017.02	0.2%
23	能源大数据	楼振飞	上海：上海科学技术出版社，2016.03	0.2%
24	MongoDB + Express + Angular + Node.js全栈开发实战派	柳伟卫	北京：电子工业出版社，2020.01	0.2%
25	中国当代教育思想宝库	陈贵	北京：人民日报出版社，2005.09	0.17%
26	银行管理与IT架构	刘士峰	上海：华东理工大学出版社，2014.02	0.17%
27	网页制作三剑客培训教程 Dreamweaver MX 2004 Fireworks MX 2004 Flash MX 2004 第4版	甘登岱	北京：机械工业出版社，2005.03	0.16%
28	ANDROID程序设计与开发 ANDROID STUDIO版	刘国柱;杜军威;QST青软实训	北京：清华大学出版社，2017.06	0.16%
29	计算机应用基础实用教程	李建刚;李强;吴鸿飞;张艳;高超;吴树锦	北京：电子工业出版社，2015.08	0.16%
30	APS.NET (C#2008) 项目开发案例教程	郎登何	北京：机械工业出版社，2010.06	0.16%
31	ASP.NET程序设计项目教程	许礼捷;陆国浩	北京：电子工业出版社，2016.01	0.15%
32	ASP.NET 2.0完全自学手册	张庆华	北京：机械工业出版社，2007.01	0.14%
33	信息化理论与指挥信息系统 中国电子学会电子系统工程分会第十八届信息化理论学术研讨会论文集	李强;林健;陈加根;滕娇春	北京：电子工业出版社，2011.10	0.13%
34	翻转课堂模式中的英语案例教学研究	孙春兰	吉林出版集团股份有限公司，2019.05	0.13%
35	企业管理与信息化融合创新	李士伟;王晓宇;董伟	北京：经济日报出版社，2017.09	0.13%
36	Visual Basic程序设计	杭国英	北京：机械工业出版社，2003.08	0.13%
37	Vue.js从入门到项目实战	刘汉伟	北京：清华大学出版社，2019	0.12%
38	ASP.NET 2.0应用开发实例	陈志坚	北京：北京理工大学出版社，2011.03	0.12%
39	全国计算机等级考试教程 二级C语言 无纸化	全国计算机等级考试教材编写组;未来教育 教育教学与研究中心	北京：人民邮电出版社，2013.02	0.12%
40	Java 2编程技术基础	谢峰;梁云娟	北京：高等教育出版社，2009.09	0.12%
41	实用数控机床技术手册	李金伴;马伟民;陆一心;张建新	北京：化学工业出版社，2007.10	0.12%
42	Java Web开发技术大全 JSP+Servlet+Struts 2+Hibernate+Spring+AJAX	李宁	北京：清华大学出版社，2009.05	0.12%
43	手机时尚理财技巧大全 网络购物、淘宝开店、银行理财、生活服务一本通	海天理财	北京：清华大学出版社，2014.08	0.11%
44	ASP程序设计 第2版	梁建武;李伟华;陈语林	北京：中国水利水电出版社，2007.07	0.11%
45	探索的足迹	张朝煌	中国评论学术出版社，2008.11	0.09%
46	大数据金融丛书 程序化交易实战 平台、策略、方法	冯永昌;景亮;易晓磊	北京：电子工业出版社，2015.11	0.09%
47	VISUAL C++从入门到实践	葛亮	北京：清华大学出版社，2009.09	0.09%
48	服务业跨国公司的贸易效应	张娟	北京：经济科学出版社，2017.11	0.08%



## 相似期刊

序号	题名	作者	出处	相似度
1	教育云环境下在线考试系统关键问题探讨与研究	易俗;孙世萌	电脑知识与技术, 2020, 第12期	0.65%
2	基于MongoDB的非关系型数据库的设计与开发	任明飞;李学军;崔蒙蒙;杨双龙;孙小奇	电脑知识与技术, 2019, 第34期	0.5%
3	基于Vue.js 框架实现移动终端数据可视化研究与实践	吴志霞;叶根梅;甘丽;周凌燕	通化师范学院学报, 2020, 第6期	0.5%
4	基于前后端分离技术的自动报表生成系统	罗帆;何晴;钟敬堂	智库时代, 2021, 第31期	0.42%
5	在线考试系统的发展现状与趋势	朱杰	无线互联科技, 2014, 第4期	0.42%
6	攀枝花学院学术成果展示平台设计与实现	刘彬	信息与电脑(理论版), 2018, 第22期	0.38%
7	用FLUENT与MONGODB构建高效海量日志采集系统	韩岩;李晓	中国新技术新产品, 2014, 第12期	0.37%
8	基于分布式操作系统的三维城市模型组织方法	张苗;李也白	计算机光盘软件与应用, 2014, 第4期	0.37%
9	基于Node.js与微信小程序的活动管理平台①	李正仁;周凯虎;王全贵;苏萌	计算机系统应用, 2019, 第12期	0.35%
10	基于AJAX和SVG的WEB远程实时监控系统	吴灿培;胡顺豪;王海航;沈先海	计算机工程与设计, 2011, 第9期	0.35%
11	基于WebSocket协议的特征匹配聊天系统	潘哲;邓芙蓉;韩明正;刘彩营	科技经济导刊, 2020, 第15期	0.34%
12	基于BS架构的门诊中医治疗系统的实现	徐璟璟	智库时代, 2020, 第41期	0.34%
13	基于SuperMap iObjects的兰州市道路地下安全预警系统的设计与实现	周运彬	城市勘测, 2018, 第A1期	0.33%
14	基于SPA技术的网络随访系统的设计与实现	陈洞天;任宇飞	中华医学图书情报杂志, 2019, 第7期	0.33%
15	基于Web的在线考试系统的设计与开发	杨静	西安文理学院学报(自然科学版), 2019, 第4期	0.32%
16	基于Web GIS的青稞种植决策系统设计与实现	谭鑫;高欢;魏茜	数字技术与应用, 2020, 第8期	0.3%
17	基于Spring Boot和Vue的火电SIS系统设计与实现④	朱锋	科技创新导报, 2019, 第26期	0.29%
18	FoxPro备注文件修复方法	瓮正科	微计算机应用, 1994, 第4期	0.24%
19	试分析在线考试系统的发展现状与研究	周黎	计算机产品与流通, 2017, 第11期	0.24%
20	专业普查大数据报送系统设计与开发	王学通;武翱;崔颖安	教育信息化论坛, 2018, 第3期	0.23%
21	Vue.js 技术在移动应用可视化平台网页设计中的应用	沈皎佶	数码世界, 2020, 第9期	0.22%
22	网络虚拟实验教学系统的设计与实现	殷磊;谢文阁;许志来;黄阿坤	软件工程, 2018, 第12期	0.21%
23	基于Node.js的高性能站群系统设计与实现	童敏;张文盛;钱立三	软件, 2018, 第8期	0.2%
24	一种自助预约购药设备设计	白永明;吴迎春;王峰云;韩辉	自动化技术与应用, 2021, 第11期	0.19%
25	基于SRC-CAD系统的水下土工布施工工艺研究	王国顺;詹文辉;胡正晓;张天兵	浙江交通职业技术学院学报, 2014, 第2期	0.18%
26	面向竞赛和教学的高校在线评测系统的改进与优化	张雨薇;李磊	教育现代化, 2019, 第30期	0.18%
27	微信校园便捷服务平台的开发与实现	刘波;黄情琴;李鑫科;刘连滴;苏泽利;余文森	现代信息科技, 2019, 第22期	0.18%
28	基于MVC模式的制造业信息管理系统开	孟君	微型电脑应用, 2021, 第3期	0.17%



	发			
29	基于Android的GDRMS移动图书馆信息服务系统设计与实现	付兵	现代情报, 2015, 第12期	0.17%
30	基于Java Web的在线考试管理系统的设计与实现	赵英萍;毛玉萃	电脑知识与技术, 2020, 第19期	0.17%
31	与医学专业相结合的数学实验教学探索	董鸽;王宏杰;陈立范;闵建中	现代商贸工业, 2021, 第32期	0.16%
32	企业可重构竞争力系统的组织特征及其评价模型	熊鹏生;柯昌英;刘虎	武汉工程大学学报, 2009, 第8期	0.16%
33	攀枝花学院图书分享系统设计	黄莎莎;程江珂	河南科技, 2018, 第13期	0.16%
34	基于NodeJS+Express框架的轻应用定制平台的设计与实现	王伶俐;张传国	计算机科学, 2017, 第A2期	0.16%
35	构建互动型的小学数学课堂教学模式	葛红艳	数学学习与研究, 2016, 第6期	0.15%
36	普通高校课程学习支持系统的设计	黄克军;龚文芳	考试周刊, 2011, 第57期	0.14%
37	政府政务微博效能评估及可视化分析	贾斯涵;王英;郝琳琳;王鑫	软件导刊, 2021, 第3期	0.14%
38	基于Vue.js构建单页面GIS应用的方法研究	邓雯婷	科技创新与应用, 2018, 第14期	0.14%
39	《天然药物学基础》微课制作初探	陈莉;刘杨	长江丛刊, 2019, 第32期	0.13%
40	基于前后端分离技术的在线购物平台研究与实现	方生;秦晓安;王睿	黄河水利职业技术学院学报, 2021, 第4期	0.13%
41	保障性住房资格管理系统设计与实现	顾梦园;叶桦;仰燕兰	计算机科学与应用, 2019, 第3期	0.13%
42	基于网络英语在线学习系统的研究与开发	常娜	网络空间安全, 2012, 第6期	0.12%
43	基于ANDROID平台的数独游戏设计与实现	郭东恩;吴刚	计算机与数字工程, 2012, 第3期	0.11%
44	应用型本科院校图书馆“创客空间”的构建及服务策略探究——以西安航空学院为例	王倩丽;杜文龙	内蒙古科技与经济, 2021, 第6期	0.11%
45	活动导学, 分层探究, 自然生成——“任意角”的教学设计与教学反思	石鹏	数学教学通讯, 2018, 第12期	0.1%
46	医院后勤设备智能化运维平台系统建设实践	赖震	江苏卫生事业管理, 2018, 第5期	0.1%
47	基于PHP的教学过程考核系统设计	黄宏志;蔡艳荣	实验科学与技术, 2018, 第5期	0.1%
48	亚龙LBD-2002数字网络语音室		中国现代教育装备, 2003, 第6期	0.1%
49	软件需求变更过程质量管理及控制初探	杨亚慧	质量与市场, 2021, 第12期	0.1%
50	医院办公室行政工作精细化管理的应用观察	徐湾湾	魅力中国, 2021, 第46期	0.1%

## 相似报纸

序号	题名	作者	出处	相似度
1	新华网开通互联网频道		通信信息报, 2007.11.23	0.1%

## 相似网络文档

序号	题名	作者	相似度
1	SAAS环境下中小微企业CRM系统的设计和实现	王钊琦	0.5%
2	基于web的高中信息技术课堂教学平台设计与实现	范忠传	0.43%
3	dreamweaver 经典网页特效500例 时间日期 固定的右上角时钟		0.41%
4	基于SSH2的资源税证明开具管理系统的设计与实现	陈毅宏	0.41%
5	喷涂生产监测数据的存储及特征分析	李博	0.38%

6	位置可视化方法及其应用研究	姚金森	0.35%
7	基于J2EE的互联网金融服务平台的设计与实现	高进	0.35%
8	直流电机调速器监控系统的设计	邵楠	0.35%
9	基于工作流的科研绩效考核系统的设计与实现	孙雨升	0.35%
10	基于对象特征的数据可视化应用系统设计及实现	常彦博	0.35%
11	智能电网中基于WebGIS的可视化分析系统研究与实现	侯雪	0.34%
12	基于Node.js的脑慢病管理系统的设计与实现	杨涛	0.32%
13	基于LAMP架构的校园微活动管理平台的设计与实现	庄欠林	0.31%
14	基于ARM的室内环境监控系统的设计与实现	何川	0.27%
15	高校贫困生评定系统的分析与设计	卢栋	0.25%
16	中小企业网络培训系统的设计与实现	李含果	0.24%
17	大数据在农业物联网中的应用研究	李春辉	0.23%
18	基于WEB的网络教学系统设计	程程	0.23%
19	基于Netty的水质监测系统的设计与实现	游文武	0.2%
20	基于HTML5技术的检察院纪检监督举报APP的设计与实现	张弘	0.2%
21	ajax原理		0.2%
22	某学院技能竞赛管理系统的设计与实现	颜玲霞	0.19%
23	秦皇岛职业技能鉴定管理系统的设计	王志强	0.18%
24	BS模式下的学生信息管理系统	李妍	0.17%
25	煤矿安全监控信息管理系统的设计与实现	焦昊	0.17%
26	基于BS模式的计算机文化基础考试系统的设计	谷涛	0.17%
27	可扩展的通用数据交换框架的研究与实现	郝颖	0.16%
28	面向精品课程系统的软件框架的设计与实现	张思洋	0.16%
29	企业内训师移动培训平台开发与应用研究	蒋家云	0.16%
30	某建筑公司工程造价管理系统的设计与实现	汤健	0.15%
31	基于Web的公安笔录制作系统的设计与实现	丁蕙	0.15%
32	材料力学在线作业与考试系统建立的理论及其程序的实现	陈定恒	0.15%
33	基于区块链的农业订单平台开发	李凤阳	0.14%
34	oracle的文件		0.14%
35	浙江工商职业技术学院学生公寓管理系统的研究与分析	屠迪峰	0.14%
36	粮库安全设备管理系统的设计与实现	汪婷	0.14%
37	基于Android的高职院校移动学习系统的设计与实现	柯善文	0.14%
38	校园线上教育平台设计与实现	李芝萱	0.13%
39	基于Vue.js框架的Web方言地图的设计与开发	毛炎	0.13%
40	福建经贸学校共青团团籍信息管理系统的研究与分析	郭勇成	0.12%
41	基于JS的社交软件的设计与实现	李盼	0.12%
42	基于群智感知的智能交通系统的研究与实现	张易丰	0.12%
43	基于网络的工厂化水培生产设施智能化选配系统的研究	张慧娟	0.11%
44	基于微服务架构的电商平台的研究与实现	熊益益	0.11%

## 全文对比

1 绪论

1.1 课题的背景和意义





C语言程序设计是很多大学专业的一门必修课程,但是这门课一般是学生的第一门关于程序设计方面的课程,在学习方面有着一定的难度,网络上有许多视频、文档类型的免费资源,但是这些资源存在着难度不一、质量参差不齐、不能很符合本校学生学习情况的问题,学生在选择时会有一定的难度。同时也有着一些供学生可以进行自我检测的题库,但是也因为不能时常进行更新造成题目难度不一或者针对性不强的问题,导致学生往往花费了大量时间,却收获甚微,呈现出事倍功半的现象。渐渐磨灭学生的学习热情,成为一个恶性循环。

再者对于老师来说,大部分学校还是采用人工出卷的方式,在筹备试卷、组织阅卷和统计学生成绩仍需要大量资源。而人工批改也是一种浪费人力和时间的方法。随着互联网技术的发展,无纸化办公成为了现代企业管理的发展趋势之一。如果在考生答题结束后,通过系统完成网上评卷,并取得相应的成绩,就可以节省大量的人力、财力、物力和时间资源。当今传统的考试方式正面临变革,先进信息技术的采用有助于加强教学管理、提高教学质量,将考试从传统模式转向具有更高管理效率的网络考试模式 [1]。

本课题计划开发一个《C语言程序设计》课程在线题库系统,可以使学生在学习,练习和测试的时候不受地域,时间的限制,学生可以通过在线测试的方式来查看自己的学习成果,并分析出自己有哪些不足。可以在系统中的留言模块寻找自己有疑问的问题的答案,可以在算法游戏模块体验到不一样的学习方式。本系统实现了自动化出题,阅卷,从工作量方面来说,一定程度上减轻了教师的负担;系统可以对学生的考试成绩进行必要的统计、对比,帮助教师了解学生在课程中对传授的知识具体掌握情况。

## 1.2 国内外研究现状

在线题库系统种类繁多,其基本的模式可以包括CS与BS两种。由于在线题库系统所表现出的强大的便捷性、超时空性,使得很多国际上大型考试都使用了网络考试系统。在我国,在线考试系统的实际应用要晚于国外,研究起步也比较晚,但是发展也比较迅速,研究和开发活动比较活跃,实际应用也比较多。但是CS结构存在开发繁琐,后期维护成本高,变更不够灵活的缺点,所以本系统采用BS结构配合近年前端流行框架Vue进行开发。

在2009-2010年期间康皓等人研发的考试系统引进了一个多媒体试题库,能够包含图像识别、动画反映、音频和视频播放等各种内容,有效地解决了学生学习枯燥乏味的问题[3]。因此在本系统中添加了“算法游戏”作品模块,来提高学生的学习积极性。并且为了方便掌握学生的学习情况,对成绩的分布,使用图表来显示,更加的明显。

## 1.3 研究内容

为了更好的实现《C语言程序设计》课程在线题库系统,本课题对在线题库系统进行了研究和分析,基于学生与教师的实用性和乐趣性进行功能模块上的需求分析与设计。需要实现对不同用户身份之间的识别,展示不同的功能模块。对用户信息,题目信息,考试信息,交流信息等进行存储与管理。为了确保考试时的时效性,为此设置了考试进行时的倒计时功能。针对平常的编程题目输入麻烦问题,本系统提供了在线的编辑器功能。经过调研,以及深刻解析系统的工作流程,对系统的功能需求进行分析,对操作流程进行简化,最终设计了本系统所拥有的各个功能模块和系统所用到的数据表结构。具体包括以下方面内容:

(1) 需求分析:本系统基于对市面现有题库系统功能的分析总结与新增乐趣学习方面需求分析,设计了各个功能模块。

(2) 系统功能模块设计:系统包括面向用户的客户端界面设计,系统后台服务器设计,用户和系统交互流程设计以及系统中各个模块的数据表设计。本系统存在三类用户,分别为学生用户,教师用户,管理员。学生用户有着单独的注册接口,登录接口,有着在线测试,成绩与试卷存储以及展示,个人信息更改,学习心得交流,算法游戏视频存储与展示,相互评分的功能模块。为教师用户在学生用户功能的基础上增加了题库试题的管理功能,学生用户整体成绩的图形化展示和表格展示。为管理员用户在教师用户的功能的基础上增加了用户管理模块,可以对所有用户信息进行管理。

## 1.4 论文结构简介

本文主要分为四部分介绍了系统从初期调查开始,经过需求分析,模块设计,最后系统得以实现。第一部分主要是调查国内外的研究现状。明确本系统实现的意义。第二部分对系统进行了可行性以及系统的主要需求分析。对各个模块的业务逻辑进行全面细致的分析,以求建造更好的系统结构。第三部分主要涉及到系统中所使用的核心技术。第四部分则对系统中数据库结构进行设计,对系统进行技术实现。展示一些用户界面。第五部分进行了系统的测试,分别从系统前后端连接,功能模块和不同设备兼容性进行测试。

## 2 可行性分析和需求分析

### 2.1 经济可行性

主要对实现系统的费用进行评估;系统客户端使用目前主流框架之一的Vue框架搭建,项目开源,有着良好的生态支持,官方提供了清晰的使用文档。服务端使用Node开发,有着和Vue相同的特点,生态良好,项目开源;在代码编写工具方面使用了VSCode,可供开发者免费使用



；只有服务器租用中需要经济支持，对于第一次租用服务器的开发者来说，费用相对较低。所以在经济方面完全满足系统开发需求。

## 2.2 技术可行性

技术可行性：技术上的可行性是能否实现该系统的关键性因素。该系统采用BS结构进行开发。数据库采用MongoDB，语言使用HTML、CSS、JavaScript、Node，在过去的学习中，个人对本次所使用的技术进行了较为全面的学习，尝试开发了博客系统和仿真网站的项目，所以在技术可行性方面满足系统开发需求。

## 2.3 需求分析

在进行系统功能需求分析时，按照各个模块的形式进行了各个模块中的功能需求分析，在进行合理分配后，使得前台用户端和后台管理端的耦合程度有效降低。保证系统平稳的运行。系统的需求结构如图2.1所示。

图2.1 系统功能需求图

### 2.3.1 注册登录需求分析

学生用户在没有账号的情况下，需要注册账号信息，在注册好用户信息后进行登录。在登录、注册时需要对用户所输入的账号和密码进行格式检测，在检测不通过时在客户端进行相应的信息提示。

教师用户账号的注册只能由管理员在系统中进行添加，防止不相关人员对题库进行错误管理。

### 2.3.2 在线测试需求分析

学生用户可以在本系统中进行自我测试，针对不同学生学习效果、进度不同的情况，需要设置不同难度比例的题目供自我测试所用。在自我测试时需要进行时间上的限制，设置时间到达试卷自我提交的功能。在用户自己进行提交时，需要检测所有题目是否全部完成，若未完成则需要对用户进行提示，显示具体哪些题目没有完成，是否在这种情况下继续提交试卷，提交试卷后系统进行答案对比，得出测试得分。

### 2.3.3 留言模块需求分析

用户需要在这里进行留言提交和展示，在输入信息并且提交后，留言展示面板需要进行更新展示。

### 2.3.4 算法游戏模块需求分析

本模块需要实现用户上传视频作品，对同学作品相互打分的功能，并且本小组成员不得对本小组作品打分，不可对同一作品进行重复打分。在上传完成后，展示界面进行实时的数据更新。

### 2.3.5 题库管理模块需求分析

在本模块中需要实现用户对题库的更新管理，可以对陈旧题目进行删除，对新题目进行添加，在添加数据方面采用多种形式上传的方式，设置excel文件和单个题目上传的模式。在删除或者添加题目成功后，数据展示页面需要进行实时的数据更新。

### 2.3.6 成绩展示模块需求分析

本模块需要实现所有学生用户的成绩展示，通过图表的形式可以让教师掌握到学生对知识的掌握情况。

### 2.3.7 用户管理模块需求分析

本模块需要实现管理员用户对学生和教师用户账号信息的管理，能对所有用户信息进行删除并且可以在这里添加教师用户的账号信息。在删除或者添加用户信息成功后需要在展示界面进行实时的更新。

### 2.3.8 个人中心模块需求分析

本模块需要实现用户对本账号的信息修改、查看历史试卷的功能。

## 2.4 功能模块逻辑分析

### 2.4.1 登录注册模块逻辑分析



在用户输入网址后跳转至登录页面，如果学生用户没有账号信息则可以进行注册账号，输入学号和密码，在输入密码时需要输入进行两次比对，前后输入的密码是否一致，防止错按键盘导致注册密码与主观意识不符的情况，然后进行格式检测，如果格式不符设定，进行提示。最后将用户信息发送至后台进行储存。在后台首先在用户的数据库中进行以学号为关键字的查找，如果本账号已存在则返回提示信息，若没有此账号，则进行存储。在有账号的情况下直接进行登录，在输入账号密码后，点击登录按钮，执行登录函数中事件，系统将账号信息发送至后台，系统在后台通过用户账号信息在user信息表中查找比对，如果密码和账号和数据库中用户信息匹配则向客户端返回正确信息，页面路由进行跳转。若密码和账号匹配不上，则返回错误信息，给用户进行提示，重新输入账号，密码信息。本模块的流程图如图2.2。

图2.2 登录模块逻辑分析图

#### 2.4.2 在线测试模块逻辑分析

在考生进入系统后，首页设置三个等级的测试难度，考生可以根据自身的学习情况来选择练习题目的难易程度，然后跳转至考生须知界面，并将试卷分为选择题，编程题两类，在点击开始考试后，进入试卷页面，显示倒计时，试题详情，此时考生可以进行作答，在规定时间内答题完毕后，点击提交按钮，系统检测用户是否答题完成有没有遗漏题目，若有则进行提示，用户选择是否继续交卷，之后系统进行评分，并且将考生试卷储存至历史试卷数据库中，以便未来进行复习，最后在页面中显示最终得分。模块逻辑分析如图2.3所示。

图2.3 在线测试模块逻辑分析图

#### 2.4.3 留言模块逻辑分析

设置留言模块的目的是在为同学们提供一个可以进行学习交流的平台，用户进入本模块后，可以在题目的输入框中输入自己要交流的问题的题目，也就是对本问题的大概说明。在详细问题的输入框中进行对问题的详细说明。然后进行提交，在下方的留言展示进行展示，一边其他用户进行回复。

或者可以在浏览其他用户信息，对其他用户所留言的问题进行解答。模块逻辑分析如图2.4所示。

图2.4 留言模块逻辑分析图

#### 2.4.4 算法游戏模块逻辑分析

在进入到算法游戏模块后，先判断该用户是否存在，若存在可以在上传视频的功能区输入自己的小组学生学号，然后对游戏视频进行上传。若这个小组的同学已经上传过视频，则不能再次进行上传，上传视频后可以对除自己所在小组的其他小组视频进行打分，在打分时首先要对该用户是否是该视频的作者小组成员进行判断，并且不能进行重复打分。最后打的分数要进行计算，得出平均值，向用户展示。模块逻辑分析如图2.5所示。

图2.5 算法游戏模块逻辑分析图

#### 2.4.5 题库管理模块逻辑分析

本模块主要为教师用户和管理员用户展示。在用户进入页面后首先需要展示本题库中已经有的一些题目，并且在验证用户权限后，可以对已有的题目进行管理。比如删除，增加。增加题目时需要为用户提供难度不同的按钮。可以使用上传excel文件，也可以进行单个题目的上传。模块逻辑分析图如图2.6所示。

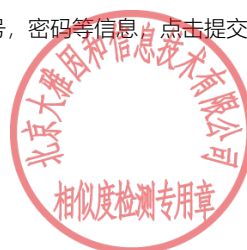
图2.6 题库管理模块逻辑分析图

#### 2.4.6 用户管理模块逻辑分析

本模块只为管理员用户展示，在页面创建之前进行身份认证，对比身份字段，如果身份字段比对不通过，则跳转至系统首页。如果身份字段比对匹配，则渲染页面。管理员用户可以看到目前系统的所有用户，并且将用户的详细信息进行展示。可以对用户进行管理，比如删除，增加。并且教师用户只能由管理员用户在此模块进行增加。需要增加教师用户时。在增加的表单中输入要增加的教师账号，密码等信息，点击提交后，由系统对教师用户进行储存。模块逻辑分析图如图2.7所示。

图2.7 用户管理模块逻辑分析图

#### 2.4.7 成绩展示模块逻辑分析





本模块为教师用户和管理员用户设置。在此模块中教师、管理员可以看到所有学生用户的成绩情况。首先整体情况以饼状图展示，同时各个试卷的详细情况以列表形式展示。可以看到试卷所属学生和答卷日期，成绩等。还可以对试卷详细答案进行查看。以便了解学生学习的具体情况。模块逻辑分析图如图2.8所示。

图2.8 成绩展示模块逻辑分析图

#### 2.4.8 个人中心模块逻辑分析

在本模块中，可以对用户的历史成绩进行查看，可以对用户的头像，昵称，密码进行修改。修改密码时需要验证原来的密码，并且在输入新密码时需要进行二次验证防止误改密码。模块逻辑分析图如图2.9所示。

图2.9 个人中心模块逻辑分析图

### 3 系统中核心技术

#### 3.1 Node.js程序

Node.js是一个运行在chromeJavaScript运行环境下（俗称GoogleV8引擎）的开发平台，用来方便快捷的创建服务器端网络应用程序。

全站开发离不开服务器的搭建，而Node.js 作为一门新出现的后台语言，可以让前端程序员经过简单的学习快速构建可伸缩的应用程序。Node中有fs模块，它补足了Node在浏览器中不能操作文件的缺点，比如使用require方法加载了fs模块后，就可以对文件进行操作。提供读，写，删除，复制等方法。并且它还提供了用来创建服务器的HTTP模块。在使用第三方的模块都是需要npm加载，并且通过require（‘模块名’）来引入。

#### 3.2 express

Express 是一个基于Node.js的服务端的开发框架，有着简洁，有着灵活的扩展特点，有着强大的路由，多模块支持的特性[4]。如path路径工具，fs操纵文件工具。有着丰富的HTTP工具和来自Connect框架的中间件，开发者可以随取随用，友好的API使得开发者可以方便快捷的创建功能强健应用。使用Express可以进行快速开发，不需要手写很多的代码，使用一行命令就可以生成一个Express的基础代码模板，并且通过其灵活的扩展机制可以很方便的加入一些其他的功能，他还提供了详细的API文档，开发人员轻松使用。

Express框架的核心特征在于：可以设置中间件来响应 HTTP 请求。可以封装一个路由文件，用来规范，定义前台的 HTTP的请求动作，更加趋向于模块化开发，使得后续的代码维护更加的容易，简单，快捷。

#### 3.3 Vue

“Vue.js”(读音vju:,类似于“view”)是一套构建用户界面的渐进式框架。与其他重量级框架不同的是，“Vue.js”采用自底向上增量开发的设计。“Vue.js”是个非常轻量级的工具。其本身具有响应式编程和组件化的特点[9]。

响应式编程简单来说，就是让数据状态和视图变化一致。在没有响应式编程的时代，在初始化界面时将数据在视图中渲染出，如果要对视图进行修改，则需要通过获取视图元素，利用相应的语法对DOM元素进行修改，保证数据和视图变化一致。但是Vue使用MVVM开发模式，使得开发者不用在关注视图和数据统一变化的问题，只需要关注数据的变化，不需要再手动获取DOM元素，让代码变的更加简洁。

组件化可以让开发者期望的代码复用得以实现。在Vue中开发者可以使用组件化这一特性，将封装的代码注册为可以复用的标签。

并且Vue也是有着丰富的生态环境。他提供了创建项目的脚手架vue-cli，不用再单独去配置webpack等配置。路由管理工具vue-router，用来实现各个页面中的跳转。全局状态变量管理工具Vuex。 HTTP请求工具axios。各种各样的UI库。使得Vue开发应变变得十分方便快捷。

#### 3.4 Ajax技术

Ajax ( Asynchronous JavaScript and XML, 异步 JavaScript和 XML)[37]最早由Jess JamesGarrett提出，区别于传统的Web应用，Ajax主要目的就是提高用户体验。

可以这样来形容，Ajax类似于一个产业链中的中间商角色，不过不同于中间商的是它并不赚取HTTP通信的差价。我们在浏览器中上网时，通过输入网址，或者通过各种各样的表单，提交或者是拿去内容时，就是在和后台的服务器进行交互。传统的web应用和服务端交互是怎样的呢？它先是由服务器端的用户来引发一条HTTP申请到达后台的服务器,而后服务器对客户端的申请做出相应的处理,之后再给服务器端回复一条

HTML的页面。在这个过程中，用户不能在客户端进行其他的操作，只能安静的等待，而且还需要宽带重新取请求整个页面。传统web应用交互逻辑图如图3.1所示。

图3.1: 传统web应用交互逻辑图

而Ajax用异步化的方式来建立起客户端和服务端的联系，在那些不需要交给服务端处理的流程直接交给Ajax来处理。经过发展，Ajax现在就像一个集合体，它集成了Css, JavaScript, XMLHttpRequest, DOM。然后通过浏览器提供的XMLHttpRequest (XHR) 对象来发出HTTP请求和接收HTTP响应。实现了web应用中界面和数据处理的分离，这样的结构有利于开发人员的分工合作，提高效率。总结来说，Ajax的优点有：

(1) 有了Ajax后，系统就可以在不刷新web应用的页面的前提下和服务器来通信，使得系统可更快的响应应用的交互操作，让客户端的等待时间减少，提高用户的使用体验。

(2) Ajax使用异步方式实现客户端和服务端的通信，在不打断用户的的操作的前提之下拥有了更加快速的响应速度，并且减少了不必要的数据传输，并节省了用户的数据流量的使用。

(3) 有了Ajax之后，以前只能是服务端负担的工作可以由后台转移到前台，利用浏览器的闲置能力来处理，以达到减轻服务端工作负担的目的。提升系统的性能。

### 3.5 MongoDB

MongoDB 是一个基于分布式，文件存储的非关系型数据库，其特点是高性能、易部署、易使用，存储数据方便[11]。

在MongoDB中文档数据以BSON格式存储，可以存储结构十分复杂的数据。但是也是因为其灵活，存储文档数据结构可以十分复杂的特性，在设计文档结构时需要谨慎的设计文档模式，以便应对数据查询困难的问题。

## 4 系统的设计与实现

### 4.1 数据库设计

经过对系统的功能需求分析，各个模块的逻辑分析，为了更好的实现系统的功能，对数据库进行了详细的设计。

#### 4.1.1 用户信息表 (user)

用户信息表用来存储用户的信息，以实现具体功能呢为目的。做出了如下设计，具体字段有xingming (用户名称)，name(账号)，pass(密码)，photo (头像路径)，student (学生用户)，teacher (教师用户)，admin (管理员)，date (创建日期)，具体结构如表4.1所示：

表4.1 用户信息数据表 (user)

序号 字段名 数据类型 默认值 是否必须

1 xingming 字符串 (String) true

2 name 字符串 (String) /files/img/photo/default.jpg true

3 pass 字符串 (String) true

4 photo 字符串 (String) true

5 student 字符串 (String) true

6 teacher 布尔值 (Boolean) false

7 admin 布尔值 (Boolean) false

8 date 数值 (Array) Date.now

#### 4.1.2 留言数据表 (textarea)



留言数据表用来存储用户的留言信息，通过数据表关联，将user用户数据表关联值留言数据表，使得每个留言有各自的留言用户信息。具体字段设置如下：title（留言标题），textarea（留言内容），date（留言时间），user（关联至具体用户），children（每个留言信息的子留言信息），子留言数组中有textarea（子留言内容），date（子留言的留言时间），user（子留言用户），replyUser（子留言用户回复对象），具体结构如表4.2，4.3所示：

表4.2 留言信息数据表（textarea）

序号 字段名 数据类型 默认值 是否必须

1 title 字符串（String） true

2 textarea 字符串（String） true

3 date 字符串（String） Date.now true

4 user 对象（Object） true

5 children 数组（Array）

表4.3 留言信息children数据表（children）

序号 字段名 数据类型 默认值 是否必须

1 textarea 字符串（String） true

2 date 字符串（String） Date.now true

3 user 对象（Object） true

4 replyUser 对象（Object） true

4.1.3 算法游戏数据表（vedio）

算法游戏数据表 用来存储用户上传的算法游戏的视频的信息，具体的字段名如下：name（小组成员），date（上传时间），newpath（视频的存储地址），arr（已打分人员信息），score（得分信息），具体结构如表4.4所示：

表4.4 算法游戏数据表（vedio）

序号 字段名 数据类型 默认值 是否必须

1 name 字符串（String） true

2 date 字符串（String） Date.now true

3 newpath 字符串（String） true

4 arr 数组（Array） true

5 score 数组（Array） true

4.1.4 题库数据表

题库的数据库表用来存储所有的题目信息，由于系统设置三类试卷难度，所以使用不同难度的数据表将题目分类，分别为简单（xeasy），适中（xmiddle），困难（xdifficult），具体字段设置如下：difficult（题目的难度），title（题目的题干），选项为A，B，C，D。answer（正确答案选项），date（系统录入本题目的时间）。编程题目的数据表设置分为besay和bdifficult两个难度。具体字段和选择题目的数据库设置类似，不设置A，B，C，D四个选项，仅设置answer字段信息。具体结构如表4.5所示：

表4.5 题库信息数据表（vedio）

序号 字段名 数据类型 默认值 是否必须

1 difficult 字符串 (String) true

2 title 字符串 (String) true

3 A 字符串 (String) true

4 B 字符串 (String) true

5 C 字符串 (String) true

6 D 字符串 (String) true

7 answer 字符串 (String) true

8 data 数值 Date.now true

#### 4.1.5 历史试卷信息数据表

历史试卷信息数据表用来存储学生用户的历史试卷信息，用来向用户展示已经做过的试卷信息，以便以后进行复习，具体字段名设置如下：list（试卷的具体信息，为一个数组），user（答题用户信息），使用数据表关联，将用户数据表和试卷信息进行关联。score（得分信息），difficult（试卷的难度分类），date（试卷存储的时间记录），具体结构表如表4.6所示：

表4.6 历史试卷信息数据表（vedio）

序号 字段名 数据类型 默认值 是否必须

1 list 数组 (Array) true

2 score 字符串 (String) true

3 user 对象 (Object) true

4 difficult 字符串 (String) true

5 date 数值 (Number) Date.now true

#### 4.1.6 系统中数据表关系

在本系统中共有五类数据表，其中的user数据表关联历史试卷数据表和留言数据表。具体关系图如图4.7所示：

图4.7 系统数据表联系图

#### 4.2 登录模块的设计和实现

在本系统中用户输入网页地址后，首先映入眼帘的是本系统的登陆界面。在设计之初，选择在本页面做出一点动态效果，以达到吸引用户的目的。本页面中主要是负责系统验证用户权限和注册学生用户。由于是客户端直面用户，所以设置了一些必要的交互效果提示。比如注册账号时，账号的格式输入不正确，在输入框失去焦点时，会进行验证，验证不通过，输入框的边框颜色变红。注册时需要输入两次密码，前后密码输入不正确，输入框会变红，并在输入框下方出现提示信息。在注册成功后，页面中会弹出用户账号注册成功的提示信息。并且页面从注册界面变为登录账号界面。在登录账号时，在前端需要进行数据格式的验证，在后端服务器中也需要进行验证，目的是防止一些通过浏览器调试工具的用户直接在控制台进行数据发送，从而跳过前端的数据验证。登陆成功后会在vuex中进行用户信息的存储，在本系统中后续需要向后台传输身份信息时都在vuex中的user信息表中拿取身份信息。

页面的星空特效使用particles插件实现，登录或者注册的表单的验证和提交使用ElementUI实现。具体的核心思想如下。

表单的数据格式设置采用正则表达式实现，正则表达式简单来说就是一串文本字符，可以使用test方法来判断字符串是否符合于开发者指定的规则。在这里指定的账号规则是12位的数字，密码的规则是6-18位字符串，字符串中可以包括数字，字母和“.,?:<>|'!"@#\$\$%^&\*()+-”符号。

密码的正则规则: `let reg = /^[w,.;<>|:"'!@#%&*()+-]{6,18}$`;

账号的正则规则: `let reg = /^[d]{12,12}$`;

具体的验证效果实现使用el-form中的validator函数。如果验证不通过进行具体信息的提示。在字符串的规则验证通过后, 点击登录按钮, 则进行数据的发送, 建立前后端的数据通信, 后端接收数据, 进行数据库查询, user信息表中是否有该账号, 以及输入密码是否和数据表中存储的字段信息相同。

具体的数据发送使用axios技术实现, 在本系统中涉及到发送或者请求数据的代码皆是通过axios实现, **请求数据方式使用Get方式, 发送数据使用Post方式。后续关于发送请求数据的过程不再赘述。**核心代码如下:

```
this.$refs[loginFrom].validate(async (valid) => {    if (valid) {        let { data } = await this.$axios({            method: "POST",            url: "/login",            data: this.loginFrom,        });        if (data.code) {            return this.$message.error(data.msg);        }        this.$message({            message: "用户登录成功",            type: "success",            duration: 1000,        });        this.updateuser(data.data)        //关闭登录页面        this.close();        this.$router.replace('/')    } else {        this.$message.error("请输入正确的用户名或密码");        return false;    } });
```

#### 4.3 题库管理模块的设计和实现

题库管理具体分为删除题目信息和**增加题目信息**。在本系统中**增加题目信息**的方式分为单个题目上传和excle文件上传。在试卷的难度处理时采用多个不同难度的数据库组合抽题的方式, 所以在上传题目的时候是需要特别标明本次上传题目的难度。选择上传单个题目时, 在第一行的有三个难度的按钮, 在上传时进行选择即可。如果选择excle文件上传文件, 则需要对excle文件按照特定格式进行编辑。方便系统解析excle文件进行解析。在解析文件过程中使用js提供的FileReader对象, 利用readAsBinaryString()方法将文件转为二进制字符串, 使用xlsx插件进行解析。通过xlsx提供的方法将二进制字符串转换为json格式数据。具体的代码如下:

```
//使用FileReader对象读取文件, 并转为二进制 let reader = new FileReader    reader.readAsBinaryString(file)    reader.onload = ev => {        resolve(ev.target.result)    } //将二进制字符串变为json数据输出。 let workbook = XLSX.read(data, { type: "binary" }), worksheet = workbook.Sheets[workbook.SheetNames[0]];    data = XLSX.utils.sheet_to_json(worksheet);
```

在使用文件上传时, 可能会因为一些错误编辑, 使得文件中的题目重复, 所以在客户端的代码运行中需要对上传的数据进行去重, 去重算法在这里不再赘述。在后台接收到数据后, 首先还是要检测本次上传的数据在数据库中是否存在, 如果存在则对重复题目不进行储存, 不存在则进行储存。不只是使用excle文件上传题目时需要进行去重, 在进行单个题目上传时, 也是需要查看数据库中是否已经存在本次上传的题目, 若存在需要给客户端用户一个提示信息。

在上传题目后, 可以在题目的展示页面进行查看, 如果上传错误或者时题库中的题目已经不适合如今的教学发展, 则可以对本题目进行删除的操作。前边已经说过MongoDB数据库的唯一值为id, 在需要进行删除时, 将id值传给后台, 系统进行数据库的查找, 删除工作。具体代码不在这里展示。题库管理界面如图4.8所示。

图4.8 题库管理界面

#### 4.4 在线测试模块的设计和实现

本页面提供的**主要功能是在线测试**, **在线测试分为简单、适中、困难三个等级**, **学生用户可以根据自身的学习程度来选择测试的难度**。在选择难度后, 首先进入考生须知界面, 在这里会看到, 试卷分为两部分, 一部分是选择题, 另一部分是编程题目。进入考试的界面后, 会有一个倒计时功能, 在规定时间内, 需要完成本模块中的试题, 如果倒计时为零, 则系统自动提交本模块的试卷。如果在倒计时没有完毕前提交试卷, 系统会检测用户有没有将本模块的试题全部完成, 如果没有完成全部试题, 系统会弹出提示选择框, 提示用户是否确认交卷, 并且会在提示信息中具体显示到底是那个题目内有完成。用户可以选择确认提交或者取消提交, 继续答题。倒计时的实现如下: 设置一个总时间的秒数值, 通过封装时间处理函数, 将时间的秒数值显示在客户端的页面中。在页面创建时启动无限循环的定时器, 定时器的规则是每隔一秒, 总时间的秒数值进行减一。然后在页面中监听时间的秒数值, 单时间的秒数值变为零时, 页面直接进行跳转。由于在进入页面时使用的replace的路由跳转方式, 所以本页面不会记录在浏览器的地址记录当中, 所以在考试时没有用户可以回退至上次的答题界面。在路由跳转的时候, 系统将本模块的答题详情记录在本次的试卷答题详情数据中, 在两个模块中的试题都完成的时候, 统一发送至后台。倒计时的具体代码实现如下:

```
Time() {    setInterval(() => {        if (this.seconds > 0) {            this.seconds -= 1;            this.countDown();        } else {            this.count = `00分00秒`;        }    }, 1000); },
```



在答题完毕后，得分的计算实现在客户端，在创建界面时从后台拿到试卷信息，答题完毕后将试卷信息中的正确答案和考生答案进行比较，从而计算出试卷得分。在线测试界面如图4.9所示。

图4.9 在线测试界面

#### 4.5 成绩展示模块的设计和实现

在本模块使用ECharts开源库，通过对数据库中的成绩数据进行分类，形成特定数据格式，通过ECharts提供的图形组件和API进行页面中图形的渲染。作用主要是通过所有的学生用户的考试成绩来大体了解学生的学习情况。主要设置两个小模块。整体成绩的饼状图展示，所有学生用户的具体成绩和试卷展示，以列表形式展示，试卷详情展示采用个人中心历史试卷同样的方式。成绩展示模块界面如图4.10所示。

图4.10 成绩展示模块饼状图界面

#### 4.6 个人中心模块的设计和实现

一个系统不可避免的就是个人信息的修改，本模块不仅提供个人信息的修改，还提供用户个人历史试卷信息的查看。

个人信息的修改包括更改用户的头像，昵称，用户账号密码。上传头像类似与上传视频。修改用户名和修改密码类似，将用户的id信息，发送到后台。根据id信息查找到用户信息，对数据对象进行更新。

历史试卷信息，每个用户在使用系统的过程中，试卷信息必定大于一。所以在设计数据表时，将一个试卷中的单个题目，答案作为一个对象，将所有的对象存储在arr数组中。展示时需要处理arr数组。个人中心界面如图4.11所示。

图4.11 个人中心界面

#### 4.7 算法游戏模块的设计和实现

设置本模块是为了增加学生的学习乐趣，为了让学生可以有一个更容易理解算法的核心思想。具体实现就是学生以小组为单位，录制关于一个算法的实现视频，进行上传，其他同学可以对该视频，进行打分。在对该模块的数据库设置时考虑到自己小组用户不能对自己小组的作品进行打分，已经打分的人员不能进行二次打分。所以增加了一个存储已经打分的用户的数组。再进行打分时，需要两次判断验证，第一次是判断打分用户是否是本小组成员，第二次是判断该用户是否已经打分。这两次判断都交予前端，可以减轻后台的负担。不论打分失败还是成功，客户端用户都可以看到提示信息。每个作品的最后得分为平均分。具体实现方式为分数数组的总和来除以分数数组的长度。

在上传作品时，需要对小组成员同时上传。在这里同样使用正则表达式进行数据规则设置。由于作品为视频，但是数据库中是不可以存储视频文件的，所以处理方式是在系统文件夹中，新建一个用来储存视频文件的vedio文件夹。数据库中存储的是文件的相对路径。上传文件使用el-upload组件进行视频文件的上传。在上传进度（UploadProgressEvent）函数中得到file.percent来处理得到文件的上传进度。在后台通过multer工具设置在磁盘中存储文件。设置文件存储地址，文件存储时的名称中添加当时时间戳，防止文件夹中的文件名称重复。具体实现代码如下。

```
const upload = multer({ storage: multer.diskStorage({ destination(req, file, cb) { cb(null, path.join(__dirname,
'.././public/files/vedio')) }, filename(req, file, cb) { let ext = '.mp4' let fileName = Date.now().toString(36) + ext req.fileName
= fileName cb(null, fileName) }}}),single("file")
```

算法游戏界面如图4.12所示。

图4.12 算法游戏界面

#### 4.8 留言模块的设计和实现

本界面为用户提供一个学习交流的平台，在这里主要是两个界面模块，一个是提交留言的模块，一个是展现所有留言的模块。可以在右边的按钮处关闭提交留言的模块。提交信息的提交信息方式和登录模块相似，采用表单提交。不同的地点是，留言需要进行留言时间的展示，所以在后台创建数据库时，记录了创建本留言的时间戳，经过封装时间展示格式函数处理，变为用户所见的时间格式。时间戳处理函数如下：

```
export default function formdate (val){ let date = new Date(val), Y = date.getFullYear(), M = date.getMonth()+1, D =
date.getDate(), hh = date.getHours(), mm = date.getMinutes(), ss = date.getSeconds(); M < 10 && (M = '0' + M) D
< 10 && (D = '0' + D) hh < 10 && (hh = '0' + hh) mm < 10 && (mm = '0' + mm) ss < 10 && (ss = '0' + ss) return `${Y}-
```

```
$(M)-${D} ${hh}:${mm}:${ss}` }
```

在留言展示模块中针对每个一级留言，还可以对每一个一级留言进行回复，为了优化用户体验，初始状态下，每个一级留言下的输入框是不可见的。为了添加可以控制单个输入框显示的按钮，在页面创建时请求到的留言对象下使用filter方法为每个留言对象添加三个属性，分别为ifShowReply（布尔类型，初始值为false），reply（字符类型），replyUser（对象类型）。ifShowReply用来控制每个留言框是否显示，reply用来储存每个留言的回复内容，replyUser用来存储回复的是谁。增加了属性之后，就可以对每个一级留言进行响应的交互操作。在提交子留言时，将数据发送至后台，系统对数据进行解析，查找数据表并进行更改。因为MongoDB数据库在生成每条数据信息时都会赋予一个唯一的id值，所以通过唯一id进行数据库的查找和更新，核心代码如下：

```
textareaDB.findByIdAndUpdate(_id,{      $push:{      children:{      textarea,      user:userId,      replyUser:replyUserId      })})
```

数据更新成功，通过客户端的消息提示给予用户提示信息，然后再对留言模块的信息重新请求，拿到数据后进行页面刷新。留言界面如图4.13所示。

图4.13 留言界面

#### 4.9 用户管理模块的设计和实现

用户信息是一个系统中相对重要的数据，所以在设计之初，便设定为只有管理员可以进行用户的信息管理，通过管理界面可以对数据库进行增删。

代码中主要通过js中使用MongoDB的一些数据库操作语句进行实现。用户管理实现界面如图4.14所示。

图4.14 用户管理界面

### 5 系统测试

#### 5.1 系统连接测试

##### 5.1.1 前后端连接测试

由于客户端和服务端使用同一个ip地址，不同端口号，所以在前后端连接需要做跨域设置，在进行连接后可以进行数据发送或者接收。在这里使用发送数据的方式测试前后端是否连接。

在登录界面输入账号，密码。在服务器端使用代码输出客户端传输的变量，查看变量是否存在。登录图如5.1，变量检验图如5.2。

图5.1 登录图

图5.2 变量验证图

综上所述，系统前后端连接正常。

##### 5.1.2 数据库连接测试

在Node中连接MongoDB数据库，可以使用输出提示信息来检验系统是否和数据库连接正常。在代码中写入连接数据库代码，在终端中启动系统，然后在终端中查看提示信息，若显示‘数据库连接成功’，则系统数据库连接成功。系统启动终端图如5.3。

图5.3 系统启动终端图

#### 5.2 系统功能测试

##### 5.2.1 输入数据检测功能测试

在本系统中用户在客户端中提交信息后，系统需要对提交的信息进行验证。比如提交数据的格式，数据长度，数据类型等。

在这里测试用例为，用户密码验证。在登录模块的设计和实现中提到了密码的验证规则，这里测试的是错误输入后，客户端的输入检测功能和提示显示。如图5.4所示。



图5.4 密码输入验证和用户提示信息图

### 5.2.2 数据提交功能测试

本系统中，数据提交采用JSON格式发送，数据提交后进行数据库的相关操作，例如存储。在这里使用留言的提交为例，在用户输入想要留言的内容后，点击提交后可以在数据库中找到相应的数据，提交成功后，留言面板中便有本次留言内容。

例如留言题目为：数据提交，留言内容为：本次数据提交仅供测试留言提交是否正确，如图5.5。

图5.5 客户端留言提交

点击提交后在数据库可视化软件中显示如图5.6所示。

图5.6 留言提交数据库显示

存入数据库成功后，重新拿取信息，在留言面板中显示，如图5.7所示。

图5.7 留言面板显示图

### 5.2.3 文件提交功能测试

本系统中在算法游戏和选择题目excel文件上传使用文件形式上传，这里使用算法游戏上传测试为例。在用户选择好上传文件后，点击上传按钮后，系统会对文件进行上传，如果上传成功，可以在后台指定的文件夹下出现本次上传文件，上传成功，客户端进行提示。上传文件界面如图5.8，文件上传成功后的界面如图5.9，文件夹下的文件如图5.10。

图5.8 上传文件界面

图5.9 文件上传成功后的界面

图5.10 文件夹下的文件

### 5.2.4 数据管理功能测试

本系统中数据管理包括题库中题目管理，用户管理。管理分为上传和更改，在这里使用上传编程题目为例。在客户端界面中输入题目，答案，选择题目难度后点击提交，如果上传成功，则在数据库中可见。上传界面图如图5.11，数据库图如图5.12。

图5.11 上传界面图

图5.12 上传后数据库图

### 5.2.5 在线测试功能测试

在进入在线测试界面后，会有题目数据的显示，倒计时功能，提交时信息提示，得分计算，在此模拟真实用户进行测试。进入首页选择试题难度，进行测试，在未做全部题目的情况下提交试卷。答题界面如图5.13，提交试卷界面如图5.14。

图5.13 答题界面

图5.14 提交试卷界面

## 5.3 兼容性测试

本系统在手机端和PC端都可以使用，在这里分别对PC端和手机端的界面显示进行测试，在这里使用首页和单个题目上传在PC端和手机端用户界面作为示例。

### 5.3.1 PC端显示测试

首页界面图如图5.15所示，单个题目上传界面如图5.16所示。

图5.15 首页界面图



图5.16 留言交流图

### 5.3.2 手机端显示测试

首页界面图如图5.17所示，单个题目上传界面如图5.18所示。

图5.17 首页界面图 图5.18 留言交流图

## 说明：

- 1.文献相似度 = 送检文章中与检测范围所有文献的相似字符数/送检文章正文字符数
- 2.去除参考文献相似度 = 送检文章中检测范围所有文献（不包括参考文献）的相似字符数/送检文章正文字符数
- 3.去除本人已发表论文相似度 = 送检文章中与检测范围所有文献（不包括自引）的相似字符数/送检论文正文字符数
- 4.单篇最大相似度：送检文章与某一文献的相似度高于全部其他文献
- 5.正文字符数：送检文章正文部分的总字符数，包括汉字、非中文字符、标点符号、阿拉伯数字（不计入空格），正文不包括关键词、目录、图片、附录、参考文献等

