毕业设计管理系统的设计与实现

摘要

本课题在云计算、大数据、分布式和微服务等技术背景下，重新改造了传统的毕业设计管理系统。不仅包含原毕业设计管理系统用户管理、论文管理、毕业选题、毕业阶段管理、文档上传、周报功能和公告等功能，还添加了毕业质量分析等大数据的特色功能。另外还将原有的单体应用改造为由多个微服务组成的分布式服务架构，大大提升了性能。

本平台采用Spring Cloud搭建，主要用到其Spring Boot、Spring Cloud Alibaba Nacos、Spring Cloud Gateway、Spring Cloud Alibaba OSS等模块。其他有MyBatis Plus、Shiro、Lombok、Hadoop等相关技术。

关键字：高校，毕业设计，微服务，分布式

Design and Implementation of Graduation Design Management System Abstract

This subject has reinvented the traditional graduation design management system under the technical background of cloud computing, big data, distributed and microservices. Not only include the original graduation design management system user management, thesis management, graduation topic selection, graduation stage management, document upload, weekly report functions and announcements, but also add the features of big data such as graduation quality analysis. In addition, the original single application is transformed into a distributed service architecture composed of multiple microservices, which greatly improves performance.

This platform is built with Spring Cloud, mainly using its Spring Boot, Spring Cloud Alibaba Nacos, Spring Cloud Gateway, Spring Cloud Alibaba OSS and other modules. Other technologies include MyBatis plus, Shiro, Lombok, and Hadoop.

**Keywords**: Colleges, Graduation Design, Microservices, Distributed

目 录

[1 引言 1](#_Toc40945888)

[1．1 课题的提出 1](#_Toc40945889)

[1．2 课题的现状极其发展 1](#_Toc40945890)

[1．3 本课题的主要工作 1](#_Toc40945891)

[2 需求分析 2](#_Toc40945892)

[2．1 任务描述 2](#_Toc40945893)

[2．2 系统功能需求 2](#_Toc40945894)

[2．3 系统数据需求 2](#_Toc40945895)

[2．4 技术支持 3](#_Toc40945896)

[3 概要分析 5](#_Toc40945897)

[3．1 目标设计 5](#_Toc40945898)

[3．2 系统总体功能设计 5](#_Toc40945899)

[3．3 数据库总体设计 6](#_Toc40945900)

[4 详细分析 9](#_Toc40945901)

[4．1 数据库设计 9](#_Toc40945902)

[4．2 功能模块设计 12](#_Toc40945903)

[5 系统实现 14](#_Toc40945904)

[6 系统调试与测试 24](#_Toc40945905)

[6．1 系统调试 24](#_Toc40945906)

[6．2 系统测试 24](#_Toc40945907)

[7 结论 26](#_Toc40945908)

[7．1 设计结论 26](#_Toc40945909)

[7．2 心得体会 26](#_Toc40945910)

[参 考 文 献 27](#_Toc40945911)

[致 谢 28](#_Toc40945912)

**1 引言**

**1．1 课题的提出**

随着计算机科学技术飞速发展，高校信息化系统建设日趋完善，许多原来靠人工完成的事务，例如高校个人信息管理、选课、图书馆服务等等现在都交给了计算机来集中管理。

传统的毕业设计管理系统作为高校信息化建设的一个重要组成部分，一段时间以来，它完成了诸如学校出题、学生选题、追踪报告等一系列传统毕业设计管理功能，帮助了数万毕业生顺利毕业。但在当前＂大数据＂与＂云计算＂等技术背景下[1]，学校、教师与学生对毕业设计管理系统提出了新的要求，而传统系统已无法满足需求。

**1．2 课题的现状极其发展**

传统的毕业设计管理系统一般基于B/S模型并采用Spring+SpringMVC+MyBatis技术或者是Struts2+SpringMVC+Hibernate技术开发[2]，这些技术已经是五年前甚至是十年前的技术，主要解决的问题还是高校对于论文选题的管理问题。而在国家加快推进教育体制改革创新的新形势下[3]，毕业设计管理系统要解决的问题不仅仅是传统的论文选题的管理，还要实现利用大数据技术对学生毕业设计进行分析，得出学生毕业设计的质量等信息。而传统的毕业设计管理系统在功能上以及性能上逐渐无法满足校、师、生的新需求，这就需要对传统的毕业设计管理系统加以升级改造，使之能够适应教育现代化要求，更好地为学校、教师、学生提供服务。

**1．3 本课题的主要工作**

本次课题设计的是毕业设计管理系统，主要工作在于从两个维度对传统毕业设计管理系统升级改造。一个维度是性能方面，利用当前比较流行的分布式、微服务等模式进行开发[4]。另一个维度是功能方面，结合大数据等相关内容，为系统添加具有大数据背景的特色功能。

**2 需求分析**

**2．1 任务描述**

本系统开发的首要任务是了解用户对于毕业设计管理的新需求，在新需求之上分析传统系统的弊端与不足。再利用当下流行技术，改造项目，优化性能。

**2．2 系统功能需求**

本系统包含传统的毕业设计管理系统已实现的功能。其主要分为三种角色：管理员、教师和学生。除修改个人信息和查看公告信息等公共功能外，每种角色对于系统的需求不同，其主要需求如下：

1. 管理员的需求

管理员负责对整个毕业设计管理系统进行全方位管理。首先是对于用户的管理，其主要功能包括：添加、删除、修改和查询用户基本信息，配置用户不同权限及访问域。其次是对于毕业设计的管理，其主要内容包括：教师添加的论文信息的管理，学生论文从选题到毕业文档及评分的管理，学生周报管理。最后是关于系统的管理，包括公告管理，修改菜单，配置云存储信息等。

1. 教师的需求

教师服务系统主要功能包括添加、删除、修改和删除论文信息，为自选论文题目的同学添加论文信息并指派教师，上传任务书，下载学生各阶段设计文档并评分，发布和编辑公告，与学生在周报进行互动。

1. 学生的需求

学生服务系统主要功能包括论文选题，上传论文各阶段文档，查看成绩等信息，与教师在周报进行互动。

**2．3 系统数据需求**

毕业设计管理系统需要涉及的表种类分为系统管理、论文管理和公告管理，这三种类表分属不同业务。其中系统管理类有用户信息表、用户角色表、菜单信息表及这三张表之间对应关系的中间表，另外还有负责登录控制的验证码表和令牌表。而论文管理类包括论文信息表，阶段表，学生阶段表，周报表。公告管理类包括公告表。

**2．4 技术支持**

1. Java

Java是一种面向对象的编程语言，它不仅融合了C语言的各种优点，而且还抛弃了C中难以理解的多重继承和指针的概念。因此，Java语言具有两个功能：功能强大且用户友好。作为静态面向对象编程语言的代表，Java语言出色地实现了面向对象理论，因此程序员可以用优雅的思维来进行复杂的编程。

Java的特点是简单，面向对象，分布，健壮性，安全性，平台独立性和可移植性，多线程，动态性等[5]。Java可以编写桌面应用程序，Web应用程序，用于分布式系统和嵌入式系统的应用程序等。

1. MySQL

MySQL是传统的关系数据库产品。其开放的体系结构使得用户具有很强的选择性。随着技术的成熟，MySQL支持更多功能，性能也在不断提高[6]。支持的平台也在增加，并且有很多人参与社区发展和维护。当前，由于MySQL的稳定功能和出色的性能，它在用户中非常受欢迎，并且只要符合GPL协议，就可以免费使用和修改。

关系数据库的特征是数据存储在不同的表中，然后这些表存储在不同的数据库中。该设计不是将所有数据存储在大型仓库中，而是加快了MySQL的读取速度，并且灵活性和可管理性也得到了显着提高。用于访问和管理MySQL数据库的最常用的标准化语言是SQL，一种结构化查询语言。SQL使在数据库中存储，更新和访问信息变得容易。

1. Spring

Spring框架是由Rod Johnson发起的开源J2EE应用程序框架。这是一个轻量级的容器，用于管理bean的生命周期。Spring解决了开发人员在J2EE开发中遇到的许多常见问题，并提供了强大的IoC，AOP，Web MVC和其他功能。Spring可以单独用于创建应用程序，也可以与Struts，Webwork，Tapestry和许多其他Web框架结合使用，并且可以与诸如Swing之类的桌面应用程序AP结合使用。因此，Spring不仅可以应用于J2EE应用程序，还可以应用于桌面应用程序和Applet。Spring框架主要由七个部分组成，分别是Spring Core，Spring AOP，Spring ORM，Spring DAO，Spring Context，Spring Web和Spring Web MVC。

1. Spring Framework

Spring Framework是Java / Java EE / .NET的多层应用程序框架。该框架基于 Expert One-on-One Java EE Design and Development文章中的代码，最初是由Rod Johnson和 Juergen Hoeller et al等人开发的。Spring框架提供了一种简单的开发方法，可以避免使用大量的属性文件和帮助程序类，这些属性文件和帮助程序类会使基础代码变得复杂和混乱。

1. Hadoop

Hadoop是由Apache Foundation开发的分布式系统基础结构[7]。用户可以开发分布式程序，而无需了解分发的基本细节。利用群集的功能进行高速计算和存储。Hadoop实现了称为HDFS的分布式文件系统（Hadoop分布式文件系统）[8]。HDFS的特点是具有很高的容错能力，旨在用于廉价硬件上。它提供高吞吐量以访问应用程序数据，并且适合具有大量数据（大量数据）的用户。HDFS放宽了POSIX的要求，并允许对文件系统中的数据进行流式访问。Hadoop框架的核心设计是：HDFS和MapReduce。HDFS提供了大量数据的存储，而MapReduce支持大量数据的计算。

1. Spring Cloud Alibaba

Spring Cloud Alibaba为分布式应用程序开发提供一站式解决方案。它包含开发分布式应用程序所需的所有组件，以便轻松地使用Spring Cloud开发应用程序。使用Spring Cloud Alibaba，只需添加一些说明和少量配置即可将Spring Cloud应用程序连接到阿里巴巴的分布式解决方案，并使用阿里巴巴中间件构建分布式应用程序系统。

1. Vue.js

Vue.js是用于创建用户界面的高级框架。与其他重量级框架不同，Vue使用自底向上的增量开发设计。Vue的核心库仅关注视图级别，非常易于学习并集成到其他库或现有项目中。另一方面，Vue能够控制使用Vue生态系统支持的单个文件组件和库开发的复杂单页应用程序。

**3 概要分析**

这个阶段的工作是划分出系统物理元素，但是每个元素仍然处于黑盒级别，具体的内容将在详细设计中。概要设计的任务还包括设计软结结构，也就是确定系统中的每个程序是由那些模块组成，以及各个模块之间的联系。

**3．1 目标设计**

本系统在功能方面分为三大模块，每一大模块都拥有自己独立的数据库。除系统管理，毕设管理，公告管理等功能模块外还设有公共模块、网关模块、第三方服务、注册中心等辅助模块。其设计的基本目标为：

1. 系统管理，包括登录，权限校验，用户管理以及菜单路由等功能。
2. 毕设管理，包括论文题目管理功能，论文与用户绑定功能，阶段管理功能，各阶段文档上传与下载功能，评分功能，用户周报功能。
3. 公告管理，包括公告的编辑与发布功能。
4. 公共模块，包含各模块公共使用的坐标应用、类、配置等信息。
5. 网关模块，作为整个系统的第一道防线，还担负有菜单路由的功能。
6. 第三方服务模块，为系统中所使用的第三方服务提供配置信息的等服务。
7. 注册和配置中心，记录了服务和服务地址的映射关系。在分布式架构中，服务会注册到这里，当服务需要调用其它服务时，就到这里找到服务的地址，进行调用。

**3．2 系统总体功能设计**

1. 登录功能

系统任何用户都使用一个登录页面进行登录，包含验证码功能。其中，系统固定一个超级管理员用户，账号为admin，其他用户只能通过管理员登录后添加账号信息才能登录。

1. 公告功能

所有用户均能看到公告，当登录角色包含编辑公告权限时可以添加、编辑、删除公告。

1. 修改个人信息功能

所有用户均可以编辑自己的部分个人信息，例如联系方式，密码。

1. 教师论文管理功能

教师可以添加、编辑、删除自己的论文信息，包括论文基本信息以及上传任务书。添加论文信息后，学生可以选题，如学生选中了自己的题目。则创建对应的学生阶段信息，其中记录着已选学生完成阶段信息，包括开题报告、中期报告、毕设说明书及其对应的成绩等信息。教师可以对每个阶段进行打分。

1. 学生论文管理系统

学生可以选择论文题目，选择后创建对应的学生阶段信息，具体内容同上。学生可以下载对应的任务书，上传阶段文档，查看教师评分。

1. 周报管理功能

学生与教师每周可以通过周报互动，答疑解惑，汇报进度。内容格式包含文字、图片、代码块、超链接等富文本内容。

**3．3 数据库总体设计**

数据库存放计算机内有组织可共享的数据集合。数据库中的数据按一定的数据模型组织、描述和储存，具有较小的冗余度、较高的数据独立性和易扩展性。

下面简单介绍一下本系统主要功能模块的主要实体信息：

用户实体信息，包含用户名、密码、姓名、手机、邮箱、创建者等信息。用户名一般为学号并作为唯一标识符。E-R图如图3.1所示。



图3.1 用户实体信息

角色实体信息，包含角色名、创建人、备注等信息。E-R图如图3.2所示。



图3.2 角色实体信息

公告实体信息，包含公告内容、发布人、编辑时间等信息。E-R图如图3.3所示。



图3.3 公告实体信息

论文实体信息，包含论文号、论文标题、教师号、教师姓名、摘要、任务书等信息。E-R图如图3.4所示。



图3.4 论文实体信息

学生阶段实体信息，包含论文ID、学号、开题报告、中期报告、毕设说明书等信息。E-R图如图3.5所示。



图3.5 学生阶段实体信息

周报实体信息，包含周报内容、学号、编辑人、编辑角色、评价、编辑时间等信息。E-R图如图3.6所示。



图3.6 周报实体信息

阶段实体信息，包含阶段名、开始时间、截至时间、阶段描述等信息。E-R图如图3.7所示。



图3.7 阶段实体信息

4 详细分析

**4．1 数据库设计**

数据库设计是一个非常重要的环节，因为所有Web项目都是基于数据库的。在给定的应用程序环境中，确定了功能之后，将通过合理的逻辑设计和有效的物理设计来构建最佳数据库表，从而在逻辑上满足用户的需求。确定逻辑结构，设计关系数据，将实体模型转换为关系模型，建立数据表，建立字段，确定数据类型的值字段和每个字段的数据长度。确定数据对象的主键。以下是毕业设计管理系统的数据库关系模式信息：

公告表用来记录公告信息，详细字段结构见表4.1。

表4.1 公告表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | id | 公告id | bigint | 20 | 是 | 是 | 否 |
| 2 | content | 公告内容 | varchar | 300 | 否 | 否 | 是 |
| 3 | editor | 发布人 | varchar | 20 | 否 | 否 | 是 |
| 4 | gmt\_create | 创建时间 | datetime | 6 | 否 | 否 | 是 |
| 5 | gmt\_modified | 修改时间 | datetime | 6 | 否 | 否 | 是 |

验证码表用来记录验证码及其对应的过期时间，详细字段结构见表4.2。

表4.2 验证码表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | uuid | uuid | char | 36 | 否 | 是 | 否 |
| 2 | code | 验证码 | varchar | 6 | 否 | 否 | 否 |
| 3 | expire\_time | 过期时间 | datetime | 6 | 否 | 否 | 是 |

菜单表用来记录所有菜单选项信息，详细字段结构见表4.3。

表4.3 菜单表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | menu\_id |  | bigint | 20 | 是 | 是 | 否 |
| 2 | parent\_id | 父菜单ID | bigint | 20 | 否 | 否 | 是 |
| 3 | name | 菜单名称 | varchar | 50 | 否 | 否 | 是 |
| 4 | url | 菜单URL | varchar | 200 | 否 | 否 | 是 |
| 5 | perms | 授权 | varchar | 500 | 否 | 否 | 是 |
| 6 | type | 类型 | int | 11 | 否 | 否 | 是 |
| 7 | icon | 菜单图标 | varchar | 50 | 否 | 否 | 是 |
| 8 | order\_num | 排序 | int | 11 | 否 | 否 | 是 |

角色表用来记录所有角色信息，详细字段结构见表4.4。

表4.4 角色表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | role\_id |  | bigint | 20 | 是 | 是 | 否 |
| 2 | role\_name | 角色名称 | varchar | 100 | 否 | 否 | 是 |
| 3 | remark | 备注 | varchar | 100 | 否 | 否 | 是 |
| 4 | create\_user\_id | 创建者ID | bigint | 20 | 否 | 否 | 是 |
| 5 | create\_time | 创建时间 | datetime | 6 | 否 | 否 | 是 |
| 6 | update\_time | 修改时间 | datetime | 6 | 否 | 否 | 是 |

角色菜单表作为角色表和菜单表的中间表，保存角色和菜单之间的对应关系，详细字段结构见表4.5。

表4.5 角色菜单表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | id |  | bigint | 20 | 是 | 是 | 否 |
| 2 | role\_id | 角色ID | bigint | 20 | 否 | 否 | 是 |
| 3 | menu\_id | 菜单ID | bigint | 20 | 否 | 否 | 是 |

用户表用来保存用户信息，其中密码经过SHA256加密方式加密，详细字段结构见表4.6。

表4.6 用户表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | user\_id | 用户id | bigint | 20 | 是 | 是 | 否 |
| 2 | username | 账号 | varchar | 20 | 否 | 否 | 否 |
| 3 | password | 密码 | varchar | 100 | 否 | 否 | 是 |
| 4 | salt | 盐 | varchar | 20 | 否 | 否 | 是 |
| 5 | mobile | 手机 | varchar | 20 | 否 | 否 | 是 |
| 6 | email | 邮箱 | varchar | 255 | 否 | 否 | 是 |
| 7 | name | 姓名 | varchar | 20 | 否 | 否 | 是 |
| 8 | status | 状态 | tinyint | 4 | 否 | 否 | 是 |
| 9 | create\_user\_id | 创建者id | bigint | 20 | 否 | 否 | 是 |
| 10 | create\_time | 创建时间 | datetime | 6 | 否 | 否 | 是 |
| 11 | update\_time | 修改时间 | datetime | 6 | 否 | 否 | 是 |

用户角色表作为用户表和角色表的中间表，保存用户和角色之间的对应关系，详细字段结构见表4.7。

表4.7 用户角色表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | id |  | bigint | 20 | 是 | 是 | 否 |
| 2 | user\_id | 用户id | bigint | 20 | 否 | 否 | 是 |
| 3 | role\_id | 角色id | bigint | 20 | 否 | 否 | 是 |

令牌表用来保存登录用户的token信息，详细字段结构见表4.8。

表4.8 令牌表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | user\_id |  | bigint | 20 | 是 | 是 | 否 |
| 2 | token | token | varchar | 100 | 否 | 否 | 否 |
| 3 | expire\_time | 过期时间 | datetime | 6 | 否 | 否 | 是 |
| 4 | update\_time | 更新时间 | datetime | 6 | 否 | 否 | 是 |

阶段表用来保存主页显示的阶段信息，详细字段结构见表4.9。

表4.9 阶段表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | id |  | bigint | 20 | 是 | 是 | 否 |
| 2 | stage\_name | 阶段名 | varchar | 20 | 否 | 否 | 是 |
| 3 | start\_time | 开始时间 | datetime | 6 | 否 | 否 | 是 |
| 4 | deadline | 截至时间 | datetime | 6 | 否 | 否 | 是 |
| 5 | description | 阶段描述 | varchar | 255 | 否 | 否 | 是 |

学生阶段信息表用来保存学生的进度和评分信息，详细字段结构见表4.10。

表4.10 学生阶段表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | id |  | bigint | 20 | 是 | 是 | 否 |
| 2 | thesis\_id | 论文id | bigint | 20 | 否 | 否 | 否 |
| 3 | student\_no | 学号 | varchar | 20 | 否 | 否 | 否 |
| 4 | proposal\_path | 开题报告路径 | varchar | 255 | 否 | 否 | 是 |
| 5 | proposal\_grade | 开题报告成绩 | varchar | 10 | 否 | 否 | 是 |
| 6 | mid\_path | 中期报告路径 | varchar | 255 | 否 | 否 | 是 |
| 7 | mid\_grade | 中期报告成绩 | varchar | 10 | 否 | 否 | 是 |
| 8 | final\_path | 毕设说明书路径 | varchar | 255 | 否 | 否 | 是 |
| 9 | final\_grade | 毕设成绩 | varchar | 10 | 否 | 否 | 是 |

论文信息表用来保存论文基本信息，详细字段结构见表4.11。

表4.11 论文信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | id | 论文id | bigint | 20 | 是 | 是 | 否 |
| 2 | thesis\_no | 论文号 | varchar | 30 | 否 | 否 | 是 |
| 3 | title | 论文标题 | varchar | 20 | 否 | 否 | 是 |
| 4 | teacher\_no | 教师编号 | varchar | 20 | 否 | 否 | 是 |
| 5 | teacher\_name | 教师姓名 | varchar | 20 | 否 | 否 | 是 |
| 6 | description | 摘要 | varchar | 300 | 否 | 否 | 是 |
| 7 | student\_no | 备选状态 | varchar | 20 | 否 | 否 | 是 |
| 8 | task\_path | 任务书路径 | varchar | 255 | 否 | 否 | 是 |
| 9 | gmt\_create | 创建时间 | datetime | 6 | 否 | 否 | 是 |
| 10 | gmt\_modified | 修改时间 | datetime | 6 | 否 | 否 | 是 |

周报表用来保存学生和教师间的周报信息，详细字段结构见表4.12。

表4.12 周报表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段 | 说明 | 数据类型 | 长度 | 自增 | 主键 | 允许空 |
| 1 | id | 周报id | bigint | 20 | 是 | 是 | 否 |
| 2 | content | 周报内容 | varchar | 800 | 否 | 否 | 是 |
| 3 | student\_no | 学生 | varchar | 20 | 否 | 否 | 是 |
| 4 | editor | 编辑人 | varchar | 20 | 否 | 否 | 是 |
| 5 | role | 角色 | varchar | 10 | 否 | 否 | 是 |
| 6 | grade | 分数 | varchar | 10 | 否 | 否 | 是 |
| 7 | gmt\_create | 创建时间 | datetime | 6 | 否 | 否 | 是 |
| 8 | gmt\_modified | 修改时间 | datetime | 6 | 否 | 否 | 是 |

**4．2 功能模块设计**

主要功能包括用户管理功能、公告功能、论文管理功能、周报功能、大数据特色功能。次要辅助功能有网关功能、第三方服务功能。以下是各功能模块详细设计：

1. 用户功能

用户功能通过三张主要表来保存用户信息、区别用户、鉴权。首先是用户表，用户表保存用户的基本信息，其中用户名字段为用户在业务上的唯一识别码，密码字段保存用户密码经过SHA256加密过后的字符串[9]，盐字段保证了无法通过彩虹表等方式破解用户密码，创建者字段可以鉴别更改用户信息的更改者，防止越权修改，其他字段保存用户基本信息。另外是角色表，角色表保存角色基本信息。最后是菜单表，菜单表保存需要的权限在内的菜单基本信息。其验证用户的方式为：用户登录后通过用户和角色的中间表判断角色，再通过角色和菜单的中间表判断该角色可使用的菜单信息，最后根据用户角色的不同显示不同的菜单信息实现鉴权。如用户强行通过改url的方式访问不属于本角色的菜单信息，菜单的鉴权信息会阻止用户访问。其中已预设分配给教师、学生、管理员的菜单页面。

1. 公告功能

公告功能仿照QQ群公告功能实现，前端设计了两种页面，一种是展示给所有人的带有界面美化的公告页面，一种是管理员的后台管理页面。

1. 论文管理功能

首先是论文添加功能，教师可以添加论文基本信息，其中，教师在添加论文时候可以指定学生学号，用来实现部分同学自选论文的功能，添加时会检查该同学是否选择过论文，如选择过则驳回添加请求，如未选择过走正常选题流程。添加好论文后需要逐个上传论文任务书。这时学生可以查看未选论文题目信息，选题后会标识对应的题目已被选择，并且创建学生阶段信息。学生即可下载任务书、上传开题报告、上传中期报告、上传毕业设计说明书。教师可随时对各阶段进行打分。

1. 周报功能

周报均保存在一张表中，通过学号进行标识，通过角色字段来区别留言用户，前端限制只有教师可以评分。本模块集成了vue-quill-editor，可以保存富文本信息。

1. 大数据特色功能

留有与Hive交互的JDBC接口，通过HQL来离线分析数据，并将数据通过Echarts展示到前台页。

1. 网关功能

网关作为该系统的第一道防线，鉴定用户访问路径，路由合法路径。

1. 第三方服务功能

第三方服务功能主要集成了阿里云提供签名功能，其原理是阿里云的获取签名后直传。用户在上传文件时会先请求第三方服务提供签名，用户在获取到签名后直接上传到阿里云OSS[10]，不必经过本应用服务器，降低了系统负担。

**5 系统实现**

下面为各模块的主要功能：

（1）登录功能

管理员、教师、学生使用统一登陆界面，页面主体为左右结构。左边为一个大banner，右边为登录凭证填写框。填写框中用户字段字段可以输入6-20个字符，密码字段可以输入8-20个字符，并且提供验证码验证功能。点击登录后会根据在创建用户时的不同身份获取到不同的菜单选项。



图5.1 登录界面

（2）管理员管理模块

可以管理整个系统，包括用户管理、角色管理、菜单管理。另外还有业务层面上的管理，包括论文管理、学生阶段管理和显示在首页的阶段管理。

主界面的阶段信息主要展示各时间段所要完成的任务，可以展示给所有用户。该设置项可在管理员-阶段管理菜单中进行设置。

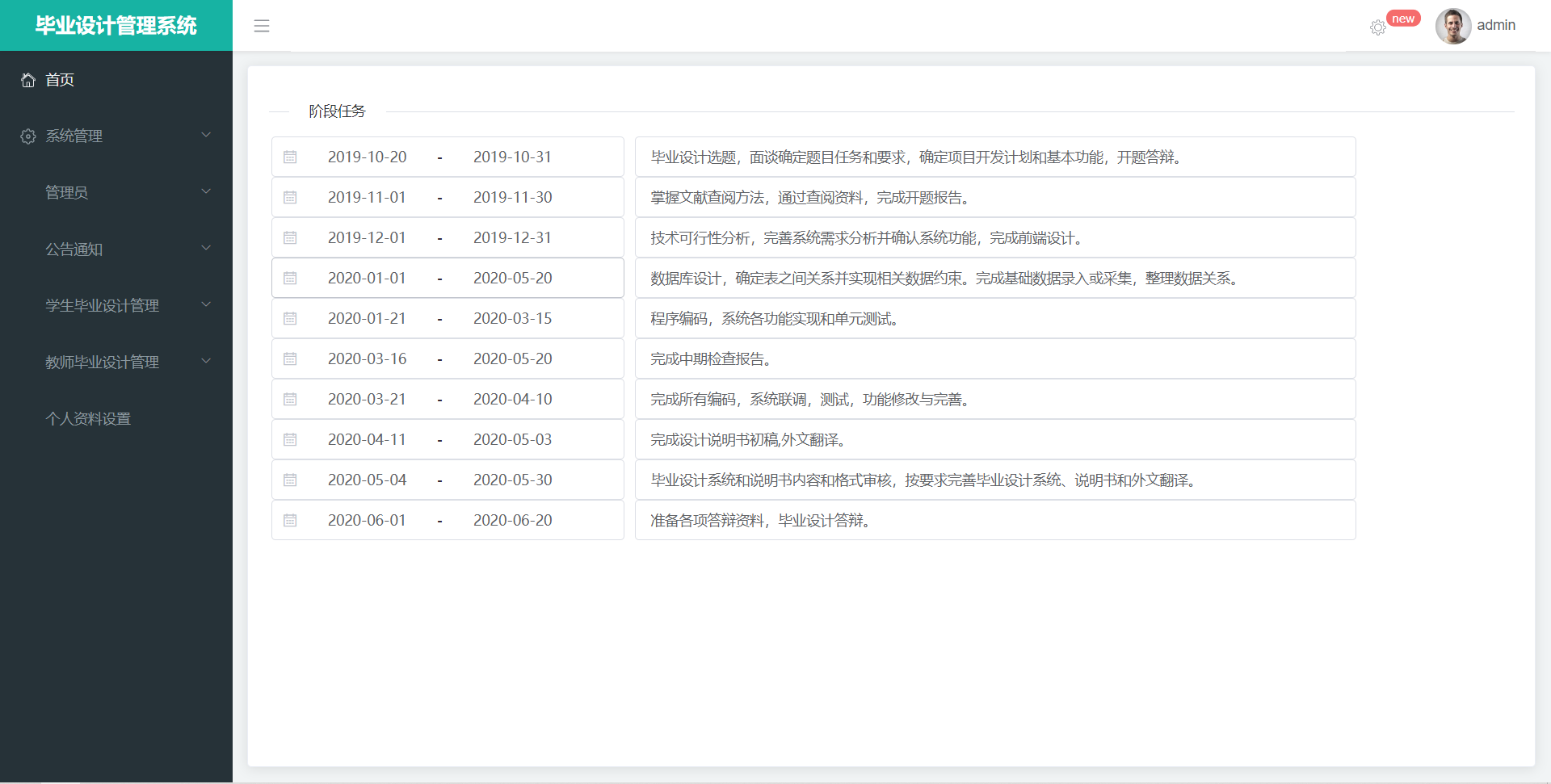


图5.2 管理员主界面

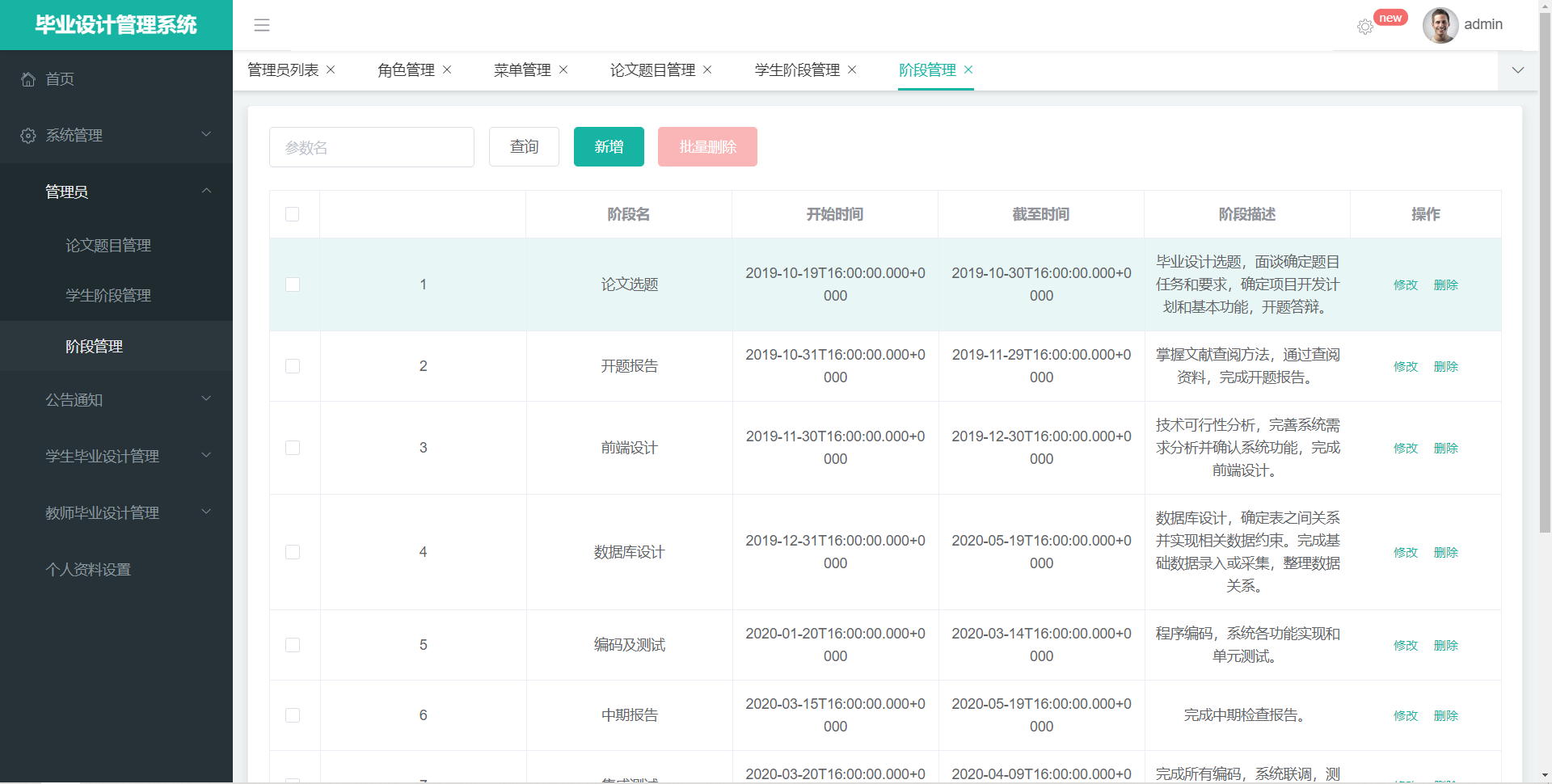


图5.3 主界面阶段管理

用户管理可以管理所有用户的信息，包括用户名、密码、邮箱、手机号、身份、是否启用等信息。其中每个用户对应一种角色，而角色可以自由定制可以查看到的菜单信息，使管理员可以很轻易地管理大批量同一角色的用户，例如学生角色可以限制他们查看学生论文管理的菜单和部分公告菜单。

图5.4 用户管理

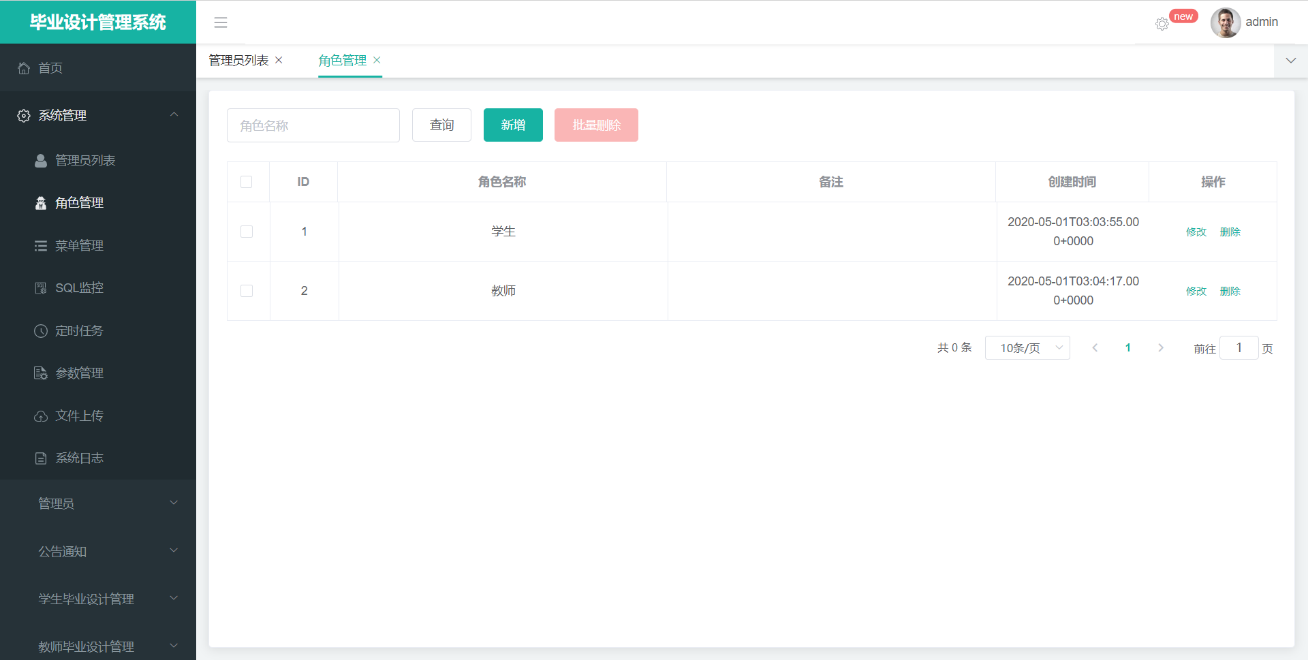


图5.5 角色管理

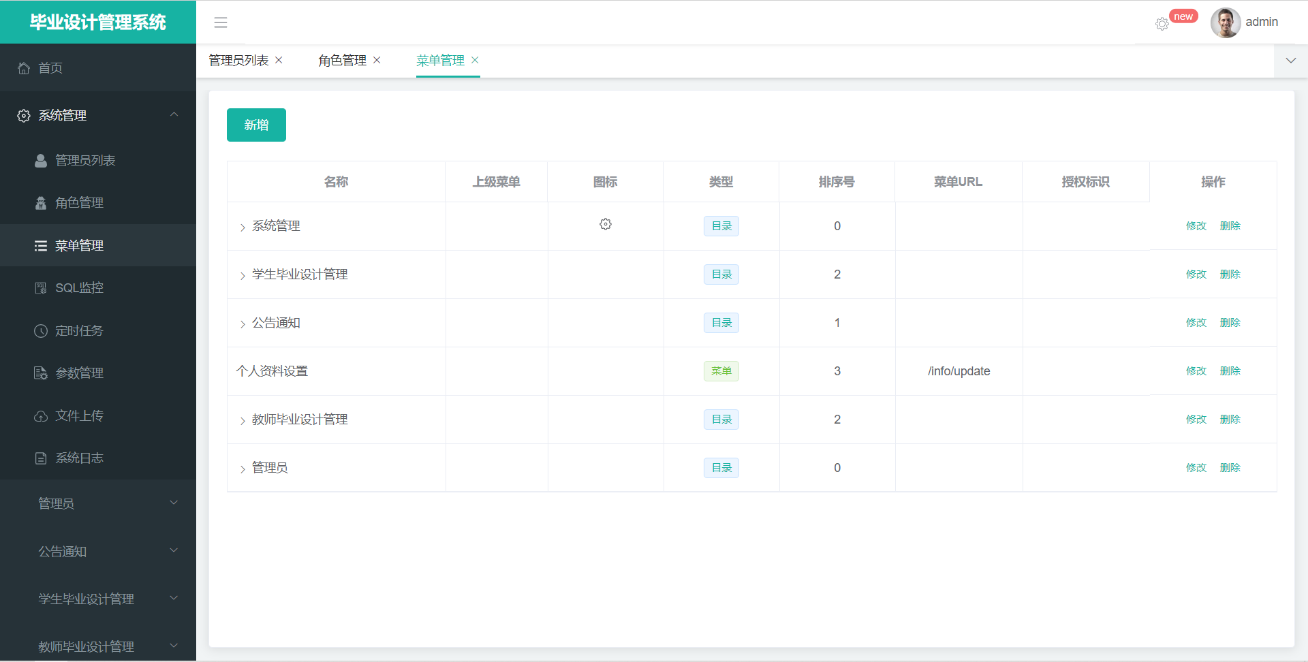


图5.6 菜单管理

接下来是业务管理部分，包括可以管理教师添加的论文，查看学生在每一阶段提交的文档信息，及每一阶段对应的教师评分。

图5.7 管理员管理论文

图5.8 管理员管理学生阶段

下面是Hadoop监控部分，主要对应于后台与Hive连接的thesis-hadoop微服务，通过向Hive里添加HQL语句，实现离线分析毕设完成情况，并将数据经由上述微服务通过Echarts图表进行展示，以下为各阶段包括开题报告、中期报告和毕设说明书的完成情况及质量情况统计。



图5.9 Hadoop监控

（3）公告模块

所有用户都应可以查看公告列表，另设有公告管理界面，管理员/教师等身份可以编辑公告。该公告列表为仿QQ群公告进行设计。

图5.10 公告列表

图5.11 公告管理

（4）教师管理模块

教师可以管理自己添加的论文，上传对应的任务书，在添加论文时，还可以指定同学已选，以方便自选课题的同学。管理选择自己论文的学生的阶段信息，下载每个同学各阶段文档，包括开题报告、中期报告、毕设说明书等。该部分的文档都会通过签名后直传的方式上传到阿里云OSS，下载也是通过阿里云OSS链接直接下载。另外教师还可以查阅和评价学生周报。

图5.12 教师论文管理

图5.13 学生阶段管理

这一部分集成了vue-quill-editor插件，可以实现富文本编辑。

图5.14 周报功能

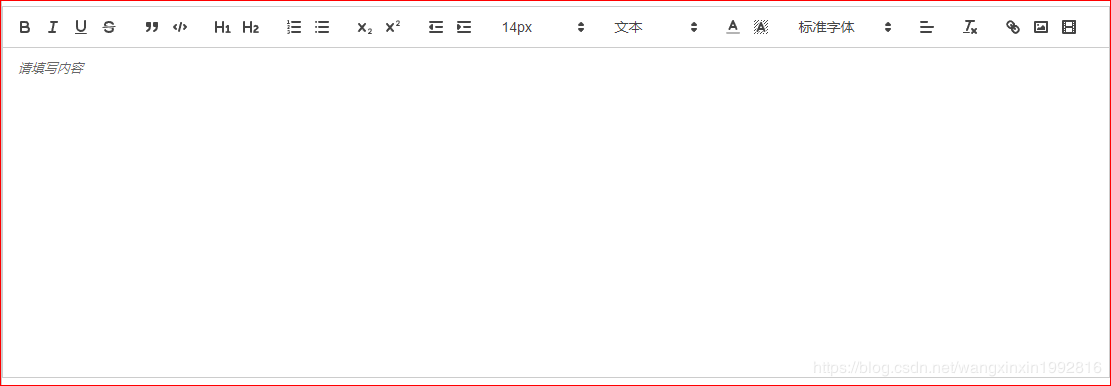


图5.15 富文本编辑器

（5）学生论文管理模块

学生可以选未被选中的题目，选题后可以在学生阶段管理页面下载任务书，上传各阶段设计文档。另外可以在周报记录页面写周报汇报每周完成情况，其中同样集成了vue-quill-editor插件。

图5.16 学生选题列表

图5.17 学生论文管理

（6）个人信息修改

所有用户均可以修改自己的部分个人信息，包括联系方式和密码，输入框均使用正则表达式做格式匹配，当点击修改时会做验证，如果格式不匹配将无法修改。其他个人信息应由管理员进行修改。

图5.18 个人信息修改

**6 系统调试与测试**

**6．1 系统调试**

在开发过程中，不可避免地会发生错误。一些语法错误和异常利用开发工具即可进行修复。一些难以发现的错误包括程序执行期间的其他隐蔽的错误，这些错误只能通过逐步调试来发现和纠正。只有经过仔细的调试，才能最终实现正常的系统运行。

**6．2 系统测试**

软件测试在软件生命周期中起着重要作用。在生命周期的每个阶段都存在测试。尽快检测出错误并修改，否则错误的延迟扩散通常会在最终测试期间造成很大的问题。测试包括代码审查，单元测试，集成测试，确认测试和系统测试[11]。

下面举例说明一些在毕业设计管理系统中涉及到的单元测试：

1. 登录模块测试实例

登录模块要求的输入条件有：用户名和密码，而且在用户名匹配之后密码进行加密，与数据库中的加密字符串也匹配之后才可以登录成功，测试用例如表6.1。

表6.1 登录模块测试实例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 等价类划分 | | 测试数据 | | 预期结果 | 实测结果 |
| 用户名 | 密码 |
| 有效等价类 | 数据库存在的匹配的用户名和密码 | admin | admin | 登录成功 | 登录成功 |
| 无效等价类 | 用户名正确密码不正确 | admin | 123456 | 登录失败，提示用户名或密码错误 | 登录失败，提示用户名或密码错误 |
| 用户名为空密码正确 |  | admin | 登录失败，提示用户名不能为空 | 登录失败，提示用户名不能为空 |
|  | 用户名正确密码为空 | admin |  | 登录失败，提示密码不能为空 | 登录失败，提示密码不能为空 |
| 用户名和密码都为空 |  |  | 登录失败，提示用户名不能为空 | 登录失败，提示用户名不能为空 |

1. 修改密码模块

用户修改密码需要提供原密码，若原密码不匹配，则个人无法修改原密码，需告知管理员修改。原密码验证成功后，新密码和重复新密码需匹配才能成功修改。如表6.2。

表6.2 修改密码模块测试实例

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 等价类划分 | | 测试数据 | | | 预期结果 | 实测结果 |
| 原密码 | 新密码 | 重复新密码 |
| 有效等价类 | 原密码正确  新密码匹配 | admin | 123456 | 123456 | 修改成功 | 修改成功 |
| 无效等价类 | 原密码不正确 | 123456 | 123456 | 123456 | 修改失败  提示原密码错误 | 修改失败  提示原密码错误 |
| 原密码正确 新密码不匹配 | admin | 123456 | 123 | 修改失败 提示新密码不匹配 | 修改失败 提示新密码不匹配 |
| 原密码为空 |  | 123456 | 123456 | 修改失败 提示原密码错误 | 修改失败 提示原密码错误 |
|  | 新密码为空 | admin |  | 123456 | 修改失败 提示新密码不匹配 | 修改失败 提示新密码不匹配 |
|  | 重复新密码为空 | admin | 123456 |  | 修改失败 提示新密码不匹配 | 修改失败 提示新密码不匹配 |

**7 结论**

**7．1 设计结论**

本次系统设计采用了软件工程开发的基本流程，包括需求分析、概念设计、详细设计、编码、单元测试、集成测试、上线发布等几个阶段。毕业设计管理系统经过测试，基本满足了所有需求，在性能方面也有较大的提升。

由于个人能力有限，开发过程和系统某些方面还存在一些不足，今后应多加改进。

**7．2 心得体会**

在这五个多月居家隔离期间，居家开发系统似乎变成了一件容易的事情，但是在看似简单的事情背后，又隐藏着许许多多的未知困难。例如开发的技术均为当下最新的技术，而最新的技术又面临着资料匮乏的局面。这就要求我们能够通过官方发布的开发手册和文档来进行开发，而很多技术诞生地都在国外，官方只提供英文文档。在查阅文档的过程中不但理解了背后的技术原理，也逐渐锻炼了自己的英文能力。

对于开发过程来说，我深深体会到了现代开发软件的简洁与高效。之前开发软件项目，大多是一个人既写前端，又写后端，其实从宏观角度来看这件事本身没有什么问题，但是从微观角度上看这两件事之间并没有一个合理的界限，导致了初期开发的系统牵一发而动全身，当我了解了这种开发模式的弊端后，我去学习了前后端分离这种开发模式，将我的开发效率提高了一个新的台阶。另外的一个问题是数据库设计不规范，导致后续推翻设计高达三、四次，具体的解决办法是我根据任务需求严格按照阿里巴巴设计规约来进行设计，最后才趋于稳定。

在开发工具的选用上，我感受到了欲先善其事，必先利其器。虽然之前一直在用IDEA开发[12]，但许多高级功能自己没有研究过，当我学会了以后，大大加速了开发效率，还有诸如GitHub[13]、Postman、Jmeter[14]、Typora、Xshell等平台和工具，都成为了我开发过程中不可或缺的好帮手。

技术选型上，可以说现在诞生了许许多多的优秀技术架构，可喜可贺的是，相当一部分都是国产产品，例如Vue.js、Hutool、MyBatis Plus[15]等等。希望我也有一天也能成为国产开发者。

最后，在这本科四年结束之际，本毕业设计为我的大学画上了一个圆满的句号。

**参 考 文 献**

1. 徐勇.大数据及云计算技术在智慧校园中的整合与应用研究[J].电脑知识与技术,2019,15(32):55-56+68.
2. 林秋虾.基于web的毕业设计管理系统的设计与实现[J].电脑知识与技术,2018,14(34):59-61.
3. 张立洁.推进我国高等教育管理体制改革创新的思考[J].中国轻工教育,2013,(06):12-14.
4. 刘嘉裕. 基于分布式微服务全链路实时监控系统设计与实现[D].北京交通大学,2018.
5. 杨凌云.试谈计算机软件Java编程特点及其技术[J].电脑编程技巧与维护,2020,(04):47-49.
6. 杨雨成,任利峰.MySQL数据库性能优化技术研究[J].科技经济导刊,2020,28(03):32.
7. 许文鹏,李胜广,赵士伟,李思,李攀.基于Hadoop框架的大数据平台探析[J].中国安防,2020,(04):38-45.
8. 金国栋,卞昊穹,陈跃国,杜小勇.HDFS存储和优化技术研究综述[J].软件学报,2020,31(01):137-161.
9. 周福伟,何永莹.采用Sha-256算法实现管理信息系统的口令管理[J].软件导刊,2007,(21):159-160.
10. 陆地,储蓄蓄,陶静联,梁海斌.基于阿里云OSS的传统文件中心改造[J].有线电视技术,2019,(06):99-101.
11. 刘佳,王丽,杨俊叶.计算机软件测试方法及应用分析[J].科技风,2020,(09):119.
12. 陈立兵.IntelliJ IDEA——开发人员利器[J].程序员,2009,(01):118-120.
13. 齐晴,曹健,刘妍岑.GitHub中软件生态系统的演化[J].计算机研究与发展,2020,57(03):513-524.
14. 冯瑶,秦洪岩,刘跃光.基于Jmeter开展接口自动化测试方法探索与实践[J].中国金融电脑,2020,(02):48-50.
15. 叶刚,王立河,王英明,谷国栋.基于MyBatis Plus的动态生成代码设计与实现[J].电脑编程技巧与维护,2019,(07):7-8.