|  |  |
| --- | --- |
| 学院标志(改) 副本 | 惠州学院  HUIZHOU UNIVERSITY |

概 要 设 计 说 明 书

|  |  |
| --- | --- |
| **项目名称** | **基于Spring boot的惠州学院学科竞赛项目管理系统** |
| **组 号** | **9** |
| **组 长** | **蔡泽林** |
| **组 员** | **卓辉亮、余泉坤、徐崇彬** |
| **专业班级** | **17软件工程3班** |
| **任课教师** | **袁晓峰教授** |

|  |  |  |
| --- | --- | --- |
| **版本号** | **版本日期** | **版本说明** |
| **1.0** | **2020.04.18** | **概要设计说明书初稿** |
|  |  |  |
|  |  |  |
|  |  |  |

目录

[1引言 3](#_Toc38137066)

[1.1编写目的 3](#_Toc38137067)

[1.2背景 3](#_Toc38137068)

[1.3定义 3](#_Toc38137069)

[1.4参考资料 4](#_Toc38137070)

[2 总体设计 4](#_Toc38137071)

[2.1 需求规定 4](#_Toc38137072)

[2.2 运行环境 5](#_Toc38137073)

[2.3 基本设计概念和处理流程 6](#_Toc38137074)

[2.3.1 基本设计概念 6](#_Toc38137075)

[2.3.2 系统处理流程 6](#_Toc38137076)

[2.4 结构 8](#_Toc38137077)

[2.4.1 身份管理子系统 9](#_Toc38137078)

[2.4.2 公告管理子系统 10](#_Toc38137079)

[2.5 功能需求与程序的关系 12](#_Toc38137080)

[2.6 人工处理过程 12](#_Toc38137081)

[2.7 尚未问决的问题 12](#_Toc38137082)

[**3**接口设计 12](#_Toc38137083)

[3.1用户接口 12](#_Toc38137084)

[3.1.1用户管理接口 12](#_Toc38137085)

[4运行设计 15](#_Toc38137086)

[4.1运行模块组合 15](#_Toc38137087)

[4.2运行控制 15](#_Toc38137088)

[4.3运行时间 15](#_Toc38137089)

[5系统数据结构设计 15](#_Toc38137090)

[5.1逻辑结构设计要点 15](#_Toc38137091)

[5.2物理结构设计要点 16](#_Toc38137092)

[5.3数据结构与程序的关系 16](#_Toc38137093)

[6系统出错处理设计 17](#_Toc38137094)

[6.1出错信息 17](#_Toc38137095)

[6.2补救措施 17](#_Toc38137096)

[6.3系统维护设计 17](#_Toc38137097)

# 1引言

## 1.1编写目的

概要设计说明又称系统设计说明。它是用来说明对程序系统的设计考虑，包括程序系统的基本处理流程、程序系统的组织结构、模块划分、功能分配、接口设计、运行设计、数据结构设计和出错处理设计等，为详细设计提供基础。

它主要为程序开发人员提供帮助，它将需求转换成为功能程序设计的架构，是程序开发人员的开发架构的主要依据。

它的预期读者包括：系统分析人员、程序员、界面工程师、测试人员。

## 1.2背景

（1）我校每年有各种类型的学科竞赛，还有各种奖项，需要登记、统计学科竞赛名称，获奖作品名称，竞赛主办，发证单位，获奖时间，所属单位，赛事类别，获奖等级，奖励级别，学科门类，获奖人数，获奖证照片，总分值等。

其中赛事类别：国际级，国家级，省部级，市级。

获奖等级：特等奖，一等奖，二等奖，三等奖，参与奖。

奖励级别：国A，国B，省A，省B

学科门类判断：是否为师范类技能竞赛

获奖学生信息（二级学院，学号，性别，学科，专业，排名）

指导老师信息（二级学院，工号，姓名，学历，职称，排名，贡献率）

（2）年末要统计成一些报表

（3）要计算指导教师的工作量。按教师填报的贡献率乘上总分计算。

## 1.3定义

|  |  |
| --- | --- |
| 项目名称 | 基于Spring Boot的惠州学院学科竞赛项目管理系统 |
| USE CASE | 用例 |
| 超级管理员 | 教务处学科竞赛管理员 |
| 管理员 | 二级学院管理员 |
| 教师 | 惠州学院教师 |
| 活动图 | 一种用来表示用户业务功能活动顺序的UML图示 |
| 序列图 | 用来表示用户在一种业务活动中，各个对象之间交互关系和交互顺序的UML图示 |
| MySQL | MySQL是一个关系型数据库管理系统，由瑞典MySQL AB 公司开发，属于 Oracle 旗下产品。MySQL 是最流行的关系型数据库管理系统之一，在 WEB 应用方面，MySQL是最好的 RDBMS (Relational Database Management System，关系数据库管理系统) 应用软件之一。 |
| JAVA | Java 是由Sun Microsystems公司于1995年5月推出的高级程序设计语言。Java可运行于多个平台，如Windows, Mac OS，及其他多种UNIX版本的系统。 |
| Sprint Boot | SpringBoot是由Pivotal团队在2013年开始研发、2014年4月发布第一个版本的全新开源的轻量级框架。它基于Spring4.0设计，不仅继承了Spring框架原有的优秀特性，而且还通过简化配置来进一步简化了Spring应用的整个搭建和开发过程。另外SpringBoot通过集成大量的框架使得依赖包的版本冲突，以及引用的不稳定性等问题得到了很好的解决 |

## 1.4参考资料

在本文档的编写过程中，参考了一下资料，如果对文档含义有模糊之处，请查阅以下相关资料：

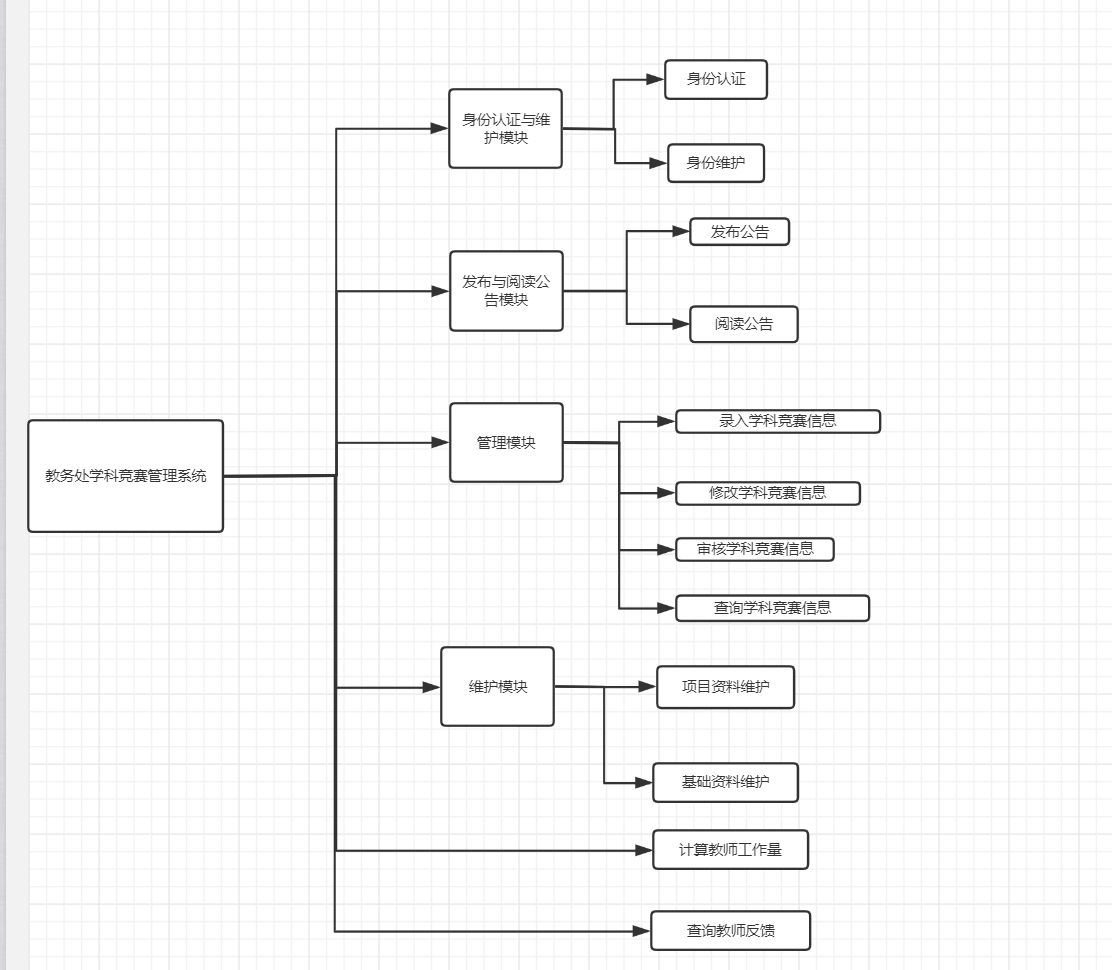
1. 《计算机程序设计艺术》——Donald Ervin Knuth

# 2 总体设计

## 2.1 需求规定

本项目通过将学科竞赛系统化，能有效查询、增删、索引学校的全部学科竞赛的信息，对于提交工作效率，节省人力资源来说，是非常重要的。下面是系统的功能：

|  |  |  |
| --- | --- | --- |
| 编号 | 模块 | 功能名称 |
| 1 | 身份认证与维护模块 | 身份认证 |
| 2 | 身份维护 |
| 3 | 发布公告与阅读公告模块 | 发布公告 |
| 4 | 阅读公告 |
| 5 | 管理模块 | 录入学科竞赛信息 |
| 6 | 查询学科竞赛信息 |
| 7 | 修改学科竞赛信息 |
| 8 | 审核学科竞赛信息 |
| 9 | 维护模块 | 项目资料维护 |
| 10 | 基础资料维护 |
| 11 |  | 计算导出指导教师工作量 |
| 12 |  | 查询教师反馈 |



## 2.2 运行环境

服务器：基于Intel构架的企业服务器

CPU：8核心cpu，两颗，主频2.0GZ以上

内存：16G及以上

存储：500G及以上

网络：10M以上

操作系统：Linux

数 据 库：MYSQL

编程语言：JAVA、HTML、JavaScript

设计工具： IEDA、StarUML

## 2.3 基本设计概念和处理流程

本项目通过将学科竞赛系统化信息化，能有效管理学校的全部学科竞赛的信息。

### 2.3.1 基本设计概念

学科竞赛系统的核心目标是为学校管理各式各样、纷繁复杂的学科竞赛项目，从而帮助学校更全面、更系统的掌握学科竞赛的信息。所以，从目标出发，系统如何更好的管理学科竞赛信息，是系统设计上，需要重点思考的基础。系统在设计概念上，将考虑一下因素：

1. 本系统在设计上，考虑到用户身份的特殊性，在身份信息上，免除身份注册的麻烦，从人事处导入所有教职工的身份信息，用户只需要通过认证即可登录使用本系统，另一方面，这也确保了用户身份的一致性，不会让非法用户进入。
2. 本系统在设计上，考虑到学科竞赛信息的多样性上，将从多方面完善学科竞赛信息的，诸如指导老师信息、学生信息、参赛队伍信息、获奖信息等等。

### 2.3.2 系统处理流程

学科竞赛信息的核心流程是教师录入、修改学科竞赛信息，二级学院领导审核信息。

录入、修改、审核学科竞赛信息流程：

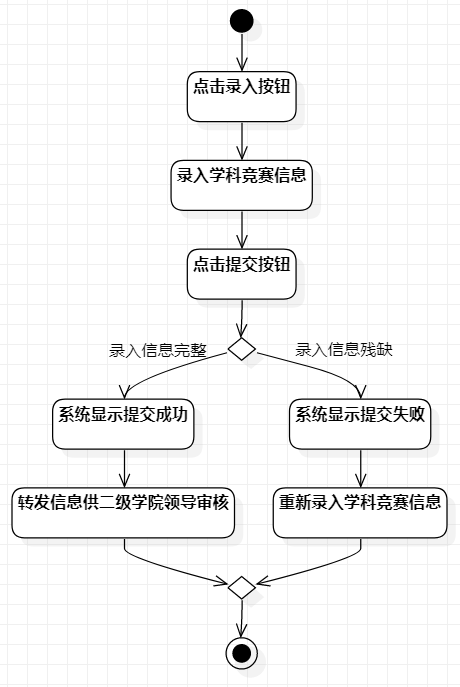


图1.教师的活动图

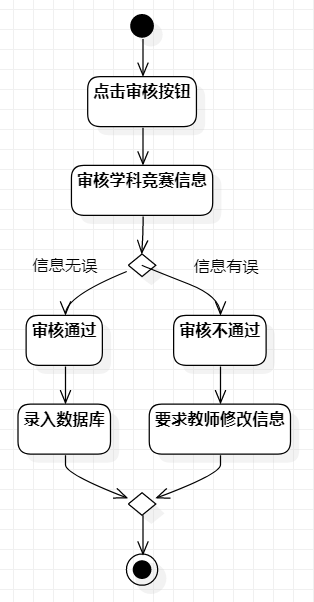
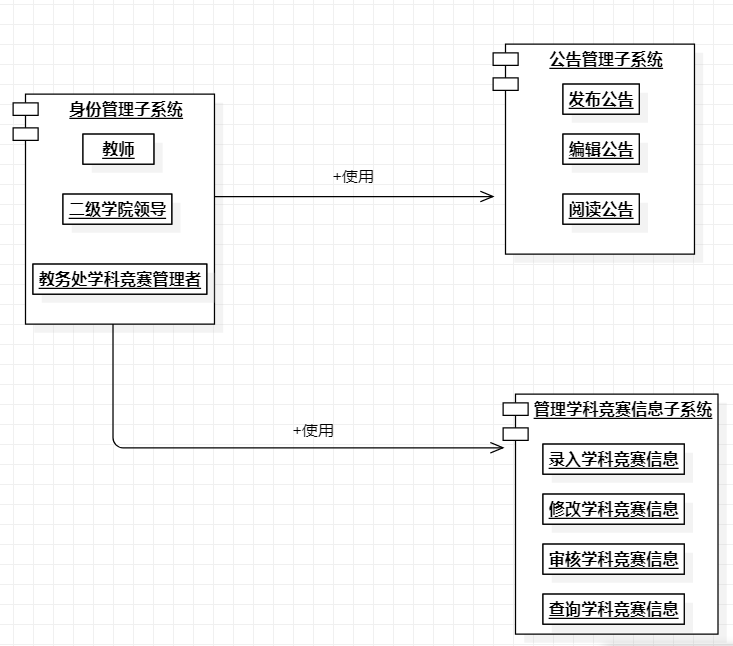


图2.二级领导的活动图

## 2.4 结构

学科竞赛项目管理系统是一个有效管理众多学科竞赛项目信息的系统，我们在用户身份管理、学科竞赛项目信息的管理上都有所针对性建构，系统分为三个子系统，它包括：身份管理子系统、公告管理子系统、学科竞赛项目信息子系统。其中，身份管理子系统包括身份的认证和身份的维护，公告管理子系统包括发布公告和阅读公告，学科竞赛项目信息管理子系统包括创建、修改、审核学科竞赛项目信息。其整体模块结构图如下：



整体模块结构图

系统在软件使用模式上，为了方便用户使用，降低用户使用成本，采用B/S架构进行构建。同时，考虑到后期扩展的可能性，在系统软件架构上，采用MVC的分层模式对系统架构进行实现，其软件架构如下

|  |  |  |  |
| --- | --- | --- | --- |
|  | 指导教师 | 二级学院主管领导 | 教务处学科竞赛管理者 |
| 功能服务层 | 1.身份认证  2.阅读公告  3.学科竞赛信息编辑（填写） | 1.身份认证  2.阅读公告  3.学科竞赛审核 | 1.身份认证  2.阅读公告  3.学科竞赛编辑  4.公告编辑  5.获奖考核  6.数据维护  7.竞赛信息导出  8.获奖考核 |
| 软件实现层 | 页面层 | HTML、JavaScript、CSS | |
| 控制层 | Controller | |
| 服务层 | Server | |
| 数据与对象关系管理层 | Mapper | |
| 基础软件层 | 应用服务器 | 阿里云服务器 | |
| 数据库服务器 | MYSQL数据库 | |
| 操作系统 | Windows、Linux | |
| 硬件服务层 | 基础网络 | 上下行带宽10M企业光纤接入及以上 | |
| 安全设备 | 防火墙一台 | |
| 物理服务器 | 应用服务器一台、数据服务器一台 | |

整个系统分为四个层级：功能服务层，软件实现层，基础软件层和硬件服务层。

其中，功能服务层通过浏览器，对各类不同用户，提供不同的功能使用。

软件实现层提供了软件的MVC实现，它在每个层级都采用标准技术来完成层次的构建，并且因为层级存在，每个层次都可以根据需求不同进行调整而不影响其它层级运行。

基础软件层为系统的运行提供基本的环境，它包括软件运行的应用服务器中间件环境、数据库环境以及操作系统环境。

硬件服务层则为系统运行提供基本的硬件基础，它包括网络环境、安全环境以及硬件服务器环境。

## 2.4.1 身份管理子系统

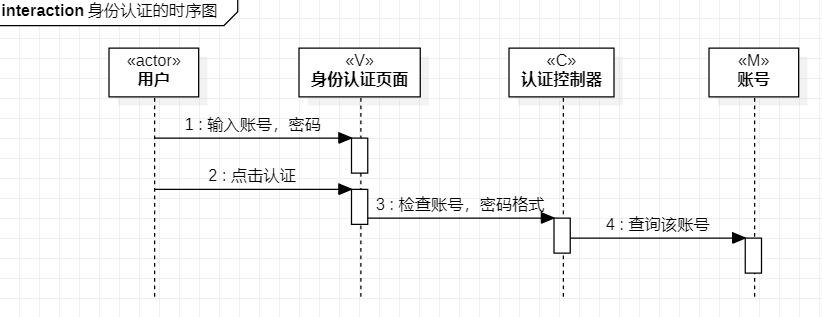
身份管理子系统能够提供两种功能：身份认证和身份维护

1. 身份认证：通过前期从导入的身份信息，用户只需认证即可登录。

其类图如下：

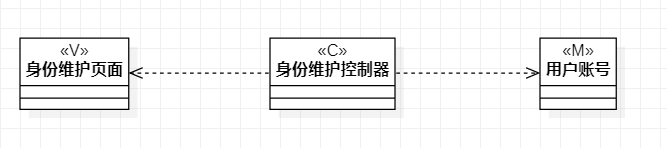


其功能设计时序图如下：

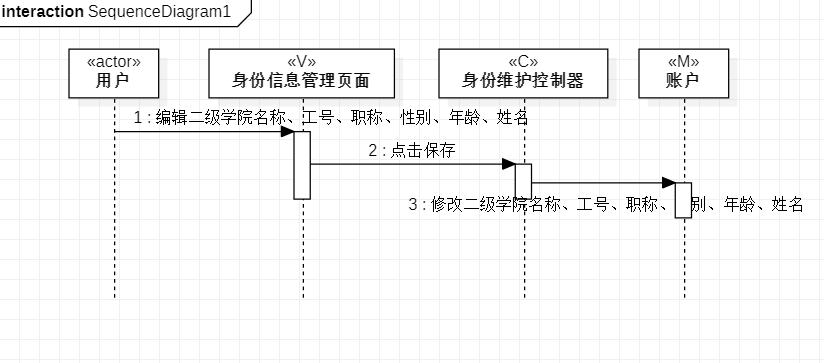


1. 身份维护：用户可以通过此功能来修改自己的信息

其类图如下：



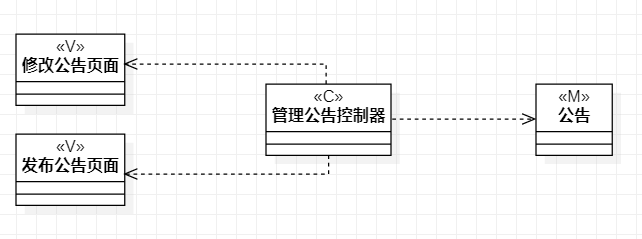
其功能设计时序图如下：



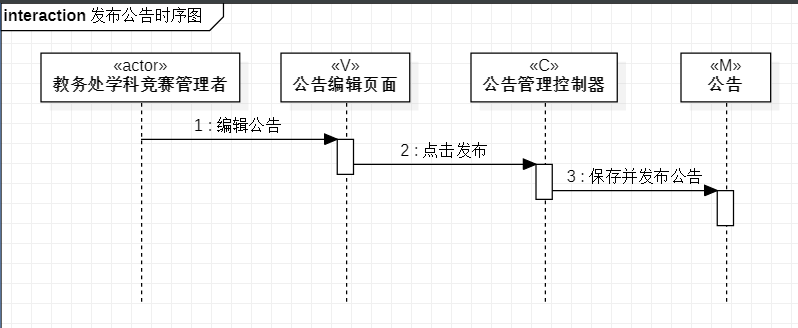
## 2.4.2 公告管理子系统

公告管理子系统包括发布公告和阅读公告

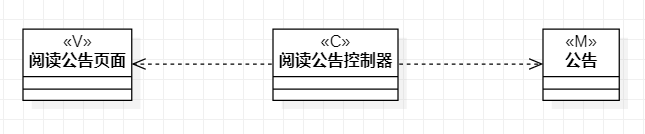
1. 发布公告包括编辑公告和发布公告，其类图如下：



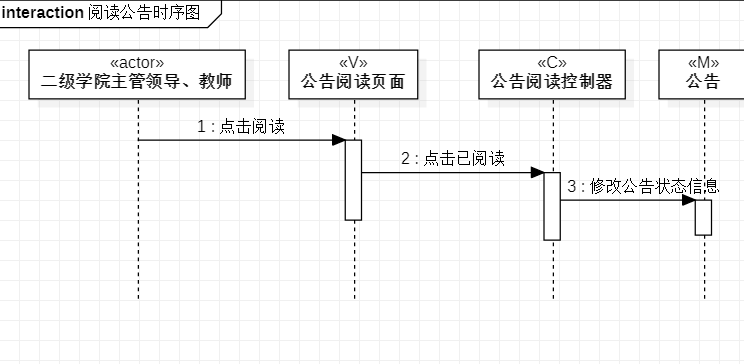
其时序图如下：



1. 阅读公告主要是来阅读公告，其类图如下：



其时序图如下：



## 2.5 功能需求与程序的关系

无

## 2.6 人工处理过程

无

## 2.7 尚未问决的问题

无

**3接口设计**

**3.1用户接口**

用户接口主要描述学科竞赛管理系统的界面原型，它是系统人机交互的接口，具体的界面如下所述：

**3.1.1用户管理接口**

1. 登录

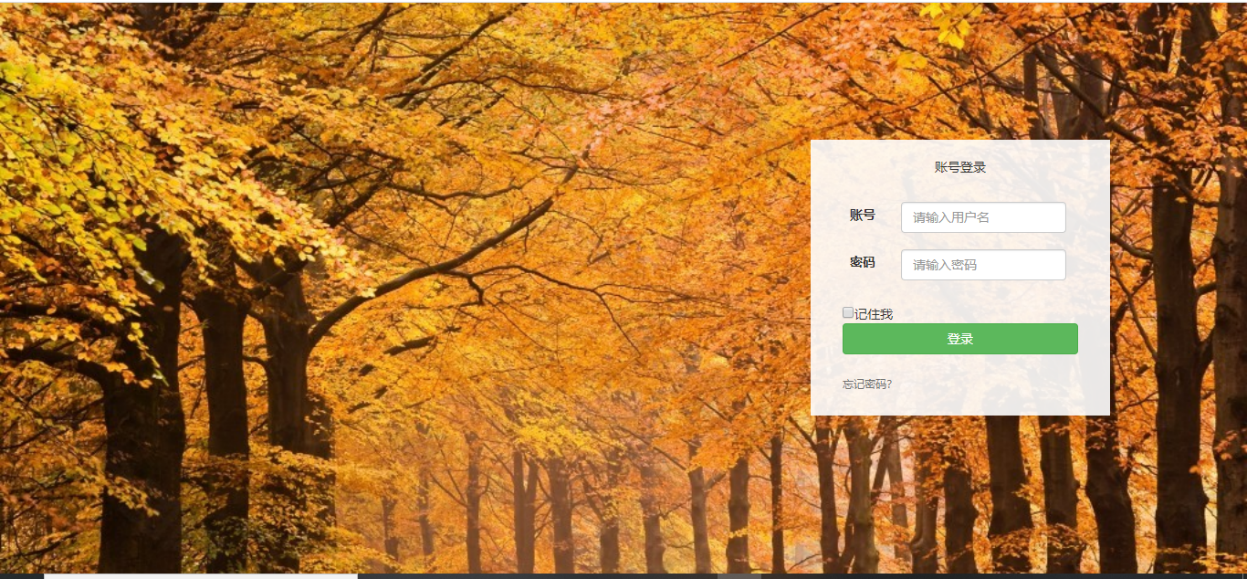


图3-1 ：登陆界面图

2. 用户个人中心（认证功能暂未编写完成）



图3-2 ：用户个人中心界面图

3.指导教师具体页面（仅有如下三项功能）

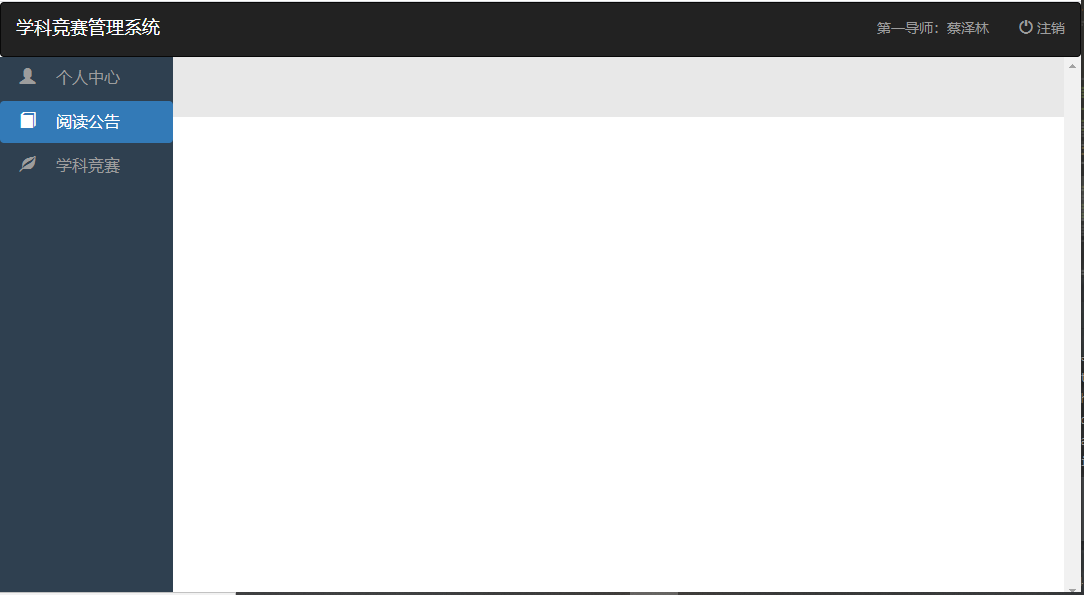


图3-3 ：指导教师界面示意图

4.二级学院管理者（仅有如下图三项功能）

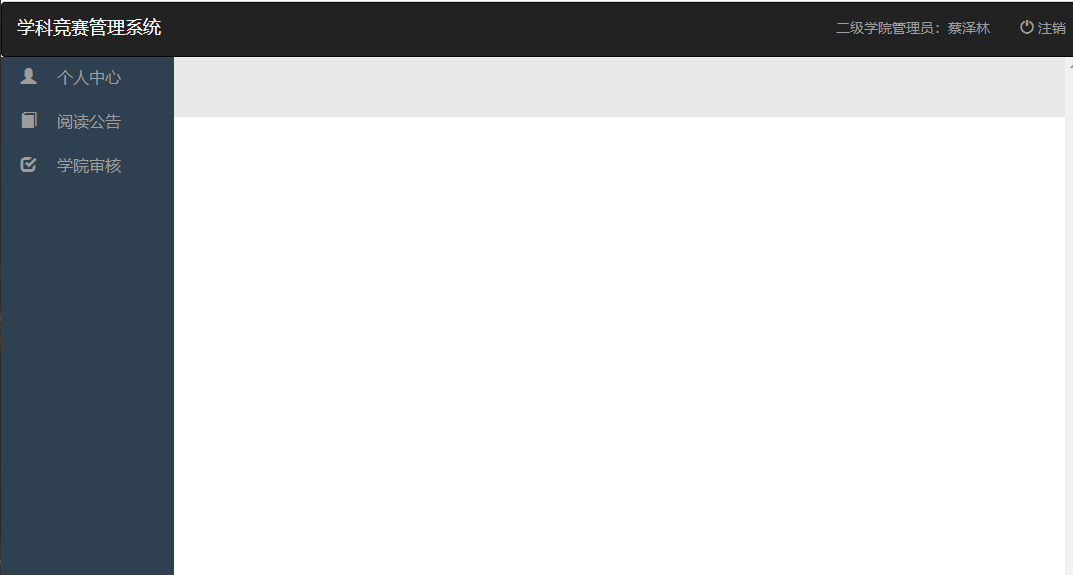


图3-4 ：二级学院管理者界面示意图

5.教务处管理员登录后的界面（有如下多项功能）

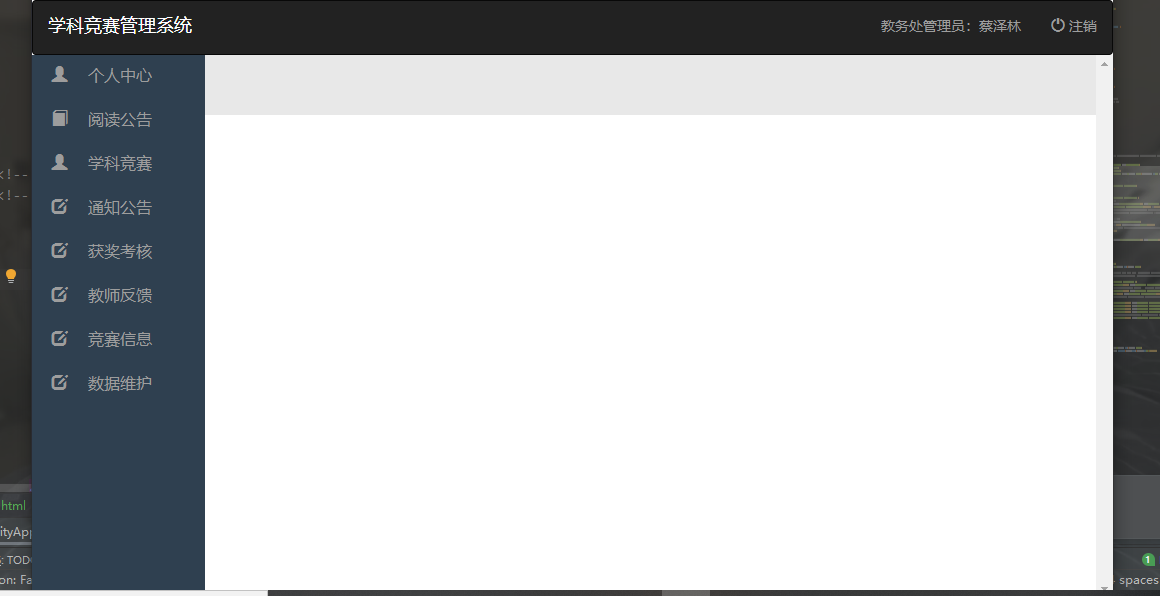
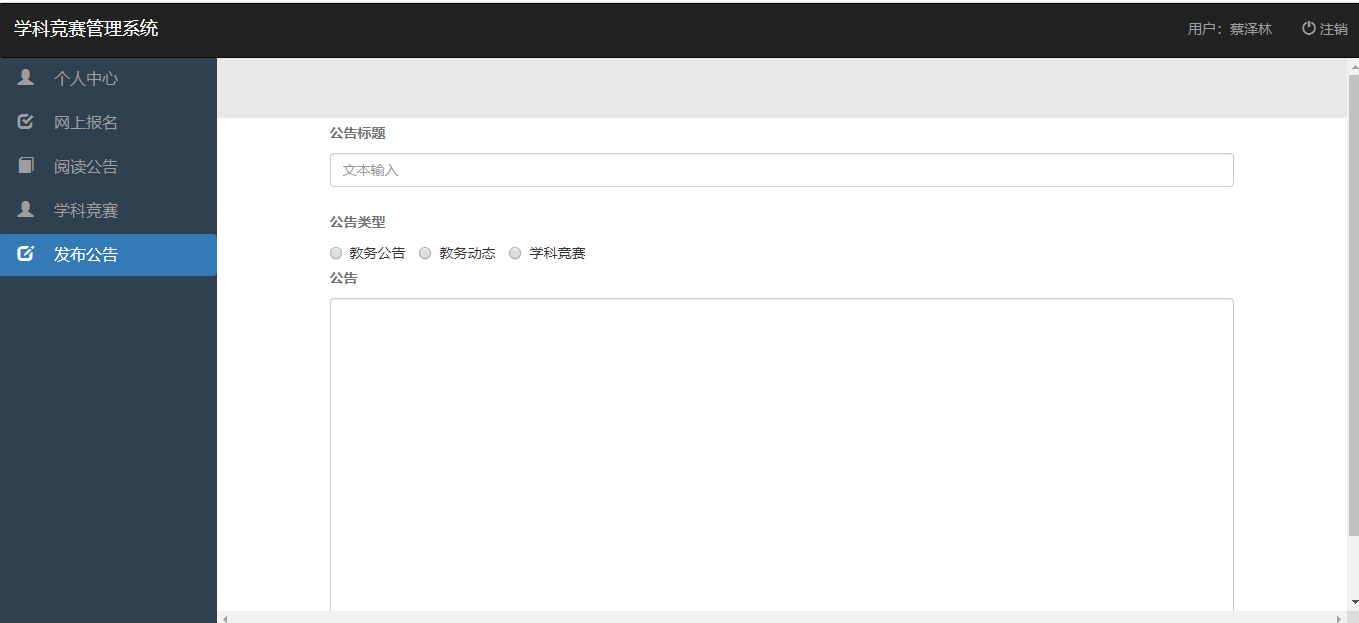
****

图3-5 ：教务处管理员示意图

6.发布通知公告功能（此功能为教务处管理员特有）：



注：其他功能页面暂未实现，后续将更新此文档。

# 4运行设计

## 4.1运行模块组合

教务处学科竞赛管理系统将运行在一个云服务器（Linux）中，云服务器预先安装JAVA，tomcat等（spring boot内嵌tomcat）。

## 4.2运行控制

暂无。

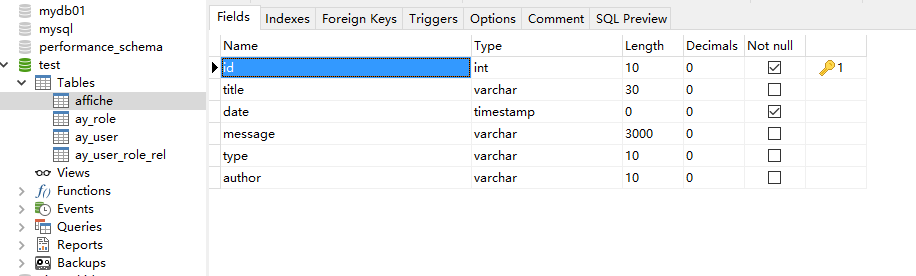
## 4.3运行时间

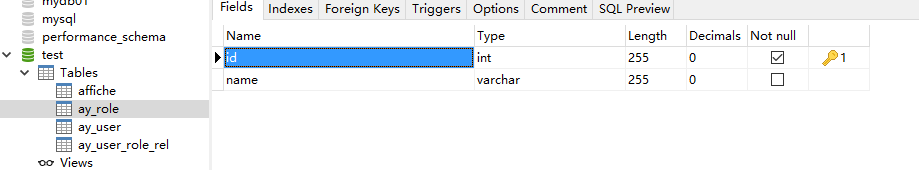
暂无。

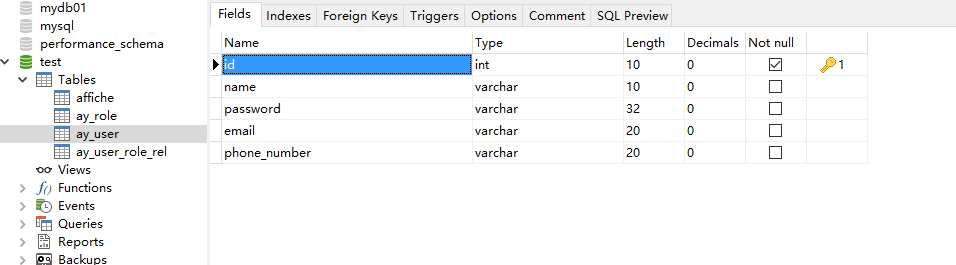
# 5系统数据结构设计

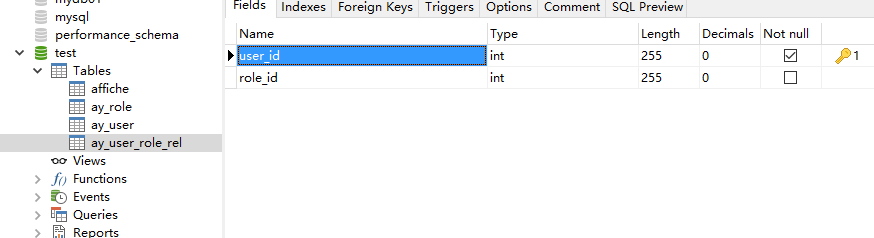
## 5.1逻辑结构设计要点

在数据结构设计中，有一些字段是预先需要定义好规则的，其规则具体如下（暂定，不代表最终版本）：









## 5.2物理结构设计要点

本系统使用mysql进行存储，而对于身份认证的用户，以及基础数据的维护，需要频繁访问数据库。为了提高系统的访问响应速度，可以建立一层缓存层，把常用的数据加载到缓存层。而redis等缓存产品对本系统过于重量级，因此本系统使用哈希表在内存中模拟一个缓存层。把经常访问的数据放到缓存层中，系统优先从缓存层取数据，取不到数据再去数据库取，同时把数据加入缓存。这样子可以有效提高系统响应速度。

## 5.3数据结构与程序的关系

具体的映射如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 数据库表 | 数据实体 | 备注 |
| 1 | **Affiche** | **Affiche.java** |  |
| 2 | **AyRole** | **AyRole.java** |  |
| 3 | **AyUser** | **AyUser.java** |  |
| 4 | **AyUserRoleRel** | **AyUserRoleRel.java** |  |

# 6系统出错处理设计

## 6.1出错信息

本系统有三类错误信息：

1. 一般性业务错误：这是系统中特定的规则，如果用户的输入不符合特定规则而产生的业务错误为此类型。
2. 系统级的错误：因为错误输入或者程序运行出现问题而引发的系统异常，都被归纳为此种错误。
3. 页面错误：一般是网络中断或程序中的BUG引起。浏览器将会提示相应的错误信息。

## 6.2补救措施

对上节所出现的三类错误，系统给出以下补救措施，包括：

* 1. 对于一般性错误，如果是用户输入问题，则主动提示用户出错的地方，帮组用户改正输入。如果是其它业务错误，则记录下错误日志，并主动提示系统管理员，然后主动告诉用户，由于故障，请用户重新进行操作。
  2. 对于系统级错误，这些错误会记录日志，并主动提示系统管理员新错误。而对于操作用户，则主动告知系统故障，请用户重新操作。
  3. 对于页面错误，由浏览器自行进行修正提示。

## 6.3系统维护设计

本系统的维护主要目标是确保系统能够7\*24小时稳定安全的运行。在此目标下，有以下措施设计来保证：

1. 提供可靠的备份机制：通过数据库软件提供的自动备份机制结合人工数据备份，形成每天增量备份，一周一次全备份的措施。并且，保留一年的数据全备份，以防止系统故障时候的恢复。
2. 每台服务器上都部署有数据库软件、应用服务器中间件以及应用，并且通过集群的方式来避免单点故障。
3. 运行日志记录在数据表中，以备系统维护人员随时查询。为了避免数据过大，日志将会一周一次备份后清空。