****

**2020届本科毕业论文（设计）**

|  |  |
| --- | --- |
| **题目：** | **基于Web的学生组织管理系统设计与实现** |

|  |  |
| --- | --- |
| **学 院 名 称** | **数学与计算机科学学院** |

|  |  |
| --- | --- |
| **专 业 名 称** | **计科（软件开发方向）** |

|  |  |
| --- | --- |
| **学　　　 号** | **1630112222** |

|  |  |
| --- | --- |
| **学 生 姓 名** | **李峥滢** |

|  |  |
| --- | --- |
| **指导教师姓名（职称）** | **李志明（助教）** |

教务处制

二〇二〇年五月

目 录

[1 引言 5](#_Toc39065356)

[1.1 背景意义 5](#_Toc39065357)

[1.2 国内外研究历史与研究现状 5](#_Toc39065358)

[1.3 论文组织结构安排 6](#_Toc39065359)

[1.4 本文研究内容 6](#_Toc39065360)

[2 系统需求分析与主要开发工具 7](#_Toc39065361)

[2.1 系统需求描述 7](#_Toc39065362)

[2.2 系统功能用例分析 7](#_Toc39065363)

[2.3 系统开发可行性分析 10](#_Toc39065364)

[2.4 相关开发技术 10](#_Toc39065365)

[2.4.1 MySQL 10](#_Toc39065366)

[2.4.2 SpringBoot+MyBatis 10](#_Toc39065367)

[2.4.3 前端框架 11](#_Toc39065368)

[2.5 本章小结 11](#_Toc39065369)

[3 系统概要设计 12](#_Toc39065370)

[3.1 总体业务框架 12](#_Toc39065371)

[3.2 浏览器/服务器 架构 12](#_Toc39065372)

[3.3 系统总体架构设计 13](#_Toc39065373)

[3.4 功能业务逻辑设计 14](#_Toc39065374)

[3.5 系统数据库设计 17](#_Toc39065375)

[3.5.1 概念结构关系 17](#_Toc39065376)

[3.5.2 核心数据表 21](#_Toc39065377)

[3.6 本章小结 24](#_Toc39065378)

[4 系统功能模块详细设计与实现 25](#_Toc39065379)

[4.1 页面登录注册功能详细设计 26](#_Toc39065380)

[4.1.1 用户注册功能 26](#_Toc39065381)

[4.1.2 用户、管理员登录功能 27](#_Toc39065382)

[4.2功能模块详细设计 28](#_Toc39065383)

[4.3 系统界面的设计与实现 31](#_Toc39065384)

[4.4 本章小结 36](#_Toc39065385)

[5 系统功能模块测试 37](#_Toc39065386)

[5.1 系统测试环境 37](#_Toc39065387)

[5.2 功能测试内容 37](#_Toc39065388)

[5.3 主要测试用例 37](#_Toc39065389)

[5.4 本章小结 40](#_Toc39065390)

[6 总结与展望 41](#_Toc39065391)

[6.1全文总结 41](#_Toc39065392)

[6.2系统存在问题和未来展望 41](#_Toc39065393)

[参考文献 42](#_Toc39065394)

[致 谢 43](#_Toc39065395)

[作者简介 44](#_Toc39065396)

[声 明 45](#_Toc39065397)

基于Web的学生组织管理系统设计与实现

计算机科学与技术（软件开发方向）专业 李峥滢

[摘要] 我国各高校学生组织众多、信息量大。目前尚未有系统性的管理，导致管理混乱、工作冲突，并且未有专属于学生组织的平台。

本文采用Java、HTML，结合Ajax异步请求技术，SpringBoot为后台框架。使用MySQL，利用Navicat Premium对数据库进行操作。该系统采用层次清晰的MVC三层架构，维护方便，从而实现本系统的科学化、规范化。

本文包括：一、前端结合Bootstrap、JQuery，使整个页面简洁大方。二、B/S架构系统，客户端仅需要下载浏览器即可使用。三、使用现阶段流行的强大框架SpringBoot开发后台逻辑事务，提升开发效率。

[关键词]学生组织；学生工作；管理系统；SpringBoot

**Design and Implementation of Student Organization Management System Based on Web**

Computer science and Technology (software development direction) LI Zheng-Ying

**Abstract:** There are many student organizations and large amounts of information in colleges and universities in China. At present, there is no systematic management, which leads to management confusion and work conflicts, and there is no dedicated platform for student organizations.

This article uses Java, HTML, combined with Ajax asynchronous request technology, SpringBoot as the background framework. Use MySQL to operate the database using Navicat Premium. The system adopts the MVC three-layer architecture with clear layers and is easy to maintain, so as to realize the scientific and standardization of the system.

This article includes: 1. The front end combines Bootstrap and JQuery to make the entire page simple and elegant. 2. B / S architecture system, the client only needs to download the browser to use. 3. Use SpringBoot, a powerful framework popular at this stage, to develop background logical transactions to improve development efficiency.

**Key words:**Student organizatio; student work; management system; SpringBoot

# 1 引言

本章主要介绍该课题的研究背景、意义以及国内外研究历史与现况。对于现阶段学生组织管理中存在的主要问题，将在本文中明确提出主要的研究内容，并给出本论文的整体组织结构布置。

## 1.1 背景意义

学生社团组织等在校园工作中起到了很大的帮助，并且提供了一个能够锻炼学生全面发展的平台。近些年，我国注重对学生的全面培育，各高校学生组织盛行起来。如学生会组织、艺术团、易班工作站、团委组织等，已成为我国高校中不可或缺的学生组织。这些组织在教师的领导下组织各类活动、协助教师完成大量的课内外校园工作，同时磨炼了意志，锻炼了人际沟通的能力。因此高校组织在校园工作方面和在全面培养人才方面都起到了巨大的作用。

而近些年来大学生数量的持续增长，也使学生组织的管理暴露出了缺陷：

首先，管理模式不完善。在数据量增大的同时，管理者仍使用旧式管理模式，即局限在Word、Excel这些文档式管理中[1]。以上情况，在较大程度上导致了高校学生组织的管理监督工作量大、工作效率水平降低。

其次，统一性弱。学生会、社团组织、艺术团等各类组织在我国高校内不断涌现。学生组织团体的性质不一、工作内容也不同。缺乏沟通交流、缺乏联系，致使组织间联系降低，从而导致在共同工作时出现工作上的疏漏或重复，诸如教室场地使用申请上的冲突、工作人员缺乏沟通导致工作的重复或遗漏等。

因此，无论是组织间还是组织内，都需要一个统一的平台，应对以上存在的问题。

## 1.2 国内外研究历史与研究现状

现今，国内各高等学校中，教务管理系统的应用相对更为成熟[2]。如：我国天津大学杨聪颖研究设计了学生信息综合管理系统[3]；韩彪研究分析的团员信息管理系统[4]；汤金等设计实现的基于Java的学生管理系统[5]。基于不同技术上：如刘丽华等刊登于通讯世界的基于SpringMVC的学生信息管理系统；王浩等刊登于中国新通信的基于NSP.NET的学生信息管理系统。然而这些现有的、针对高校的实用型系统都是侧重于对学生信息的管理，尚未有供各大学生组织统一管理、统一沟通的系统。

王明贵在教育文化论坛中发表的《分权与边界:基于对美国高校中层管理考察的探讨——以美国罗格斯大学为例》中提：学生社团作为许多学生学习和成长的重要平台，在罗格斯大学就有700多个，并且其管理部门和管理者的权力和责任黑白分明[6]。点到了国内外的两种不同现状，表露出国内高校学生组织中存在的问题：管理不统一、职责混乱。

## 1.3 论文组织结构安排

论文结构安排如下：

第一章：引言。包括本课题研究背景、意义以及当前阶段国内外的研究情况。

第二章：开发工具和相关理论技术概述。介绍实现本系统过程中需要使用到的开发工具、系统配置环境以及技术。

第三章：系统需求分析。主要针对总体业务、需求以及可行性进行描述。

第四章：系统功能模块概要设计。主要论述框架设计和功能模块的大体框架。

第五章：系统功能模块详细设计与实现。介绍各模块的核心实现方法设计以及界面展示。

第六章：系统功能模块测试。针对该系统测试环境给出测试用例。

第七章：总结。对本文工作做出总结，并对此系统在未来的研究工作做进一步的展望。

## 1.4 本文研究内容

为了解决组织管理混乱问题、组织沟通问题，使用SpringBoot框架开发学生组织管理系统。本文在开发管理系统的功能上主要包含以下三个主要方面的功能：管理员的三级联动管理、公共场地申请、沟通交流功能。基于B/S架构，前端运用Html语言和JavaScript语言，采用Bootstrap技术使得页面简洁大方，便于系统的操作使用。

# 2 系统需求分析与主要开发工具

需求分析是软件工程中关键性的一步。调查并全面性地解读该系统的需求，描述系统目标、模块功能、可行性、数据保障性等要求。并且将在本章介绍主要开发技术。

## 2.1 系统需求描述

管理员对于学生组织的管理，包括对于组织、部门和用户三个层级的CRUD操作；而普通用户仅有查看的权限。

文件管理最主要是用于全部用户的存取便利。管理员有权对所有文件进行CRUD操作；而普通用户仅可查看、上传文件并且有权删除个人上传的文件，但是没有权限删除他人上传的文件。

场地申请管理：普通用户代表组织部门向管理员提出场地申请，可以查看本部门场地申请记录；而管理员不仅可以进行场地申请和查看场地申请记录，还可以删去场地申请记录。

管理人员可以通过公告通知管理进行公告的发布、删除，而普通用户只能够查阅公告。

普通用户在工作圈模块中可以进行活动的发布和删除个人发布的内容。同时可以进行评论，并且删除个人的评论。而管理员则负责对活动和评论的管理，即有权删除他人发布的活动和评论。

## 2.2 系统功能用例分析

本小结将对该系统功能模块进行用例分析。依据总体业务描述，系统功能模块用例图如图3-2所示：

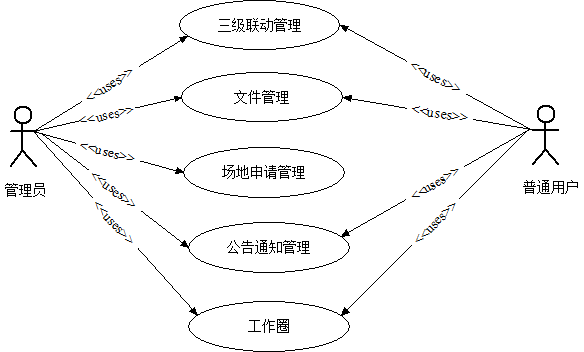


图3-2 系统用例图

各功能模块的用例图如图3-3至图3-7所示：

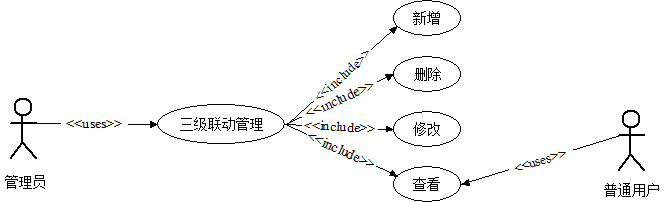


图3-3三级联动管理用例图

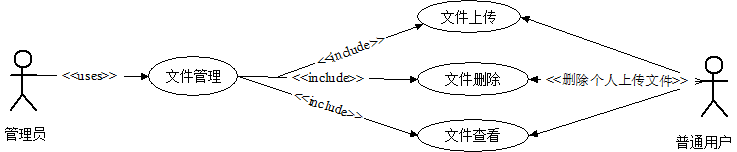


图3-4 文件管理用例图

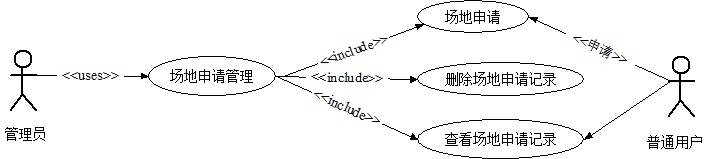


图3-5 场地申请管理用例图

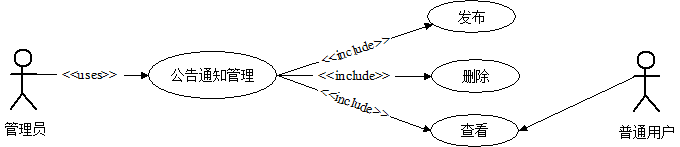


图3-6 公告通知管理用例图

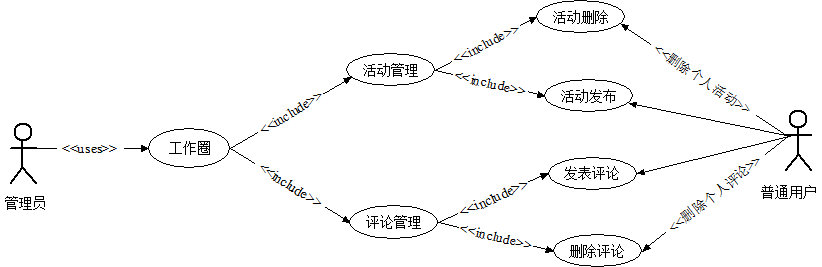


图3-7 工作圈用例图

1、三级联动管理模块

该模块是用于管理组织、组织下的部门、部门下的成员的三级联动管理，教师、管理员拥有CRUD权限，而普通用户只有查看功能。

2、文件管理模块

该模块中，教师、管理员可管理所有文件，包括增加、删除、查找，而普通成员可以上传文件和删除个人上传的文件。在查询过程中，可以通过选择不同组织、不同部门的分类方式查看文件。

3、场地申请管理模块

该模块主要用于各组织需要使用公共场地时，可以选择场地、使用时间、使用目的等信息，向教师发送申请。教师可以依据申请理由是否合理，选择是否核准。

4、公告通知模块

公告、通知众多，需要有恰当的管理方式。教师、管理员可以对公告通知进行CRUD操作，而普通用户可根据组织、部门的划分来查看公告和通知。

5、工作圈模块

该模块用于各组织、部门在工作上的交流讨论，借鉴微信App的朋友圈功能，通过发布活动内容，包括文案、配图，用户可以评论点赞，互相交流。

## 2.3 系统开发可行性分析

利用Java 开发语言基于SpringBoot框架在Eclipse开发平台上开发本系统。使用MySQL数据库，并借助数据库可视化工具Navicat Premium对其进行管理。B/S架构下使用Google Chrome。首先，借助开发工具Eclipse，利用现有主流SpringBoot框架、Maven等技术，可以完成80%的需求。其次，采用MVC框架（Model View Controller）提升系统可维护性，前端JavaScript、JQuery、Ajax、Bootstrap框架，从而提高项目质量。最后，本系统的开发绝对不会侵犯他人利益，更不会违背国家法律法规政策。

## 2.4 相关开发技术

### 2.4.1 MySQL

MySQL数据库采用SQL语言，这使得对于数据库的操作更加通俗易懂。它的灵活性表现于将数据信息保存在各不相同的表中，而不是一个大仓库中。除此以外，它开放源码、体积较小而且成本偏低。

### 2.4.2 SpringBoot+MyBatis

SpringBoot能够简化新Spring应用的初始创建以及简化整个项目的开发过程。

Spring是针对bean的生命周期管理的轻量级容器框架。其主要组成： Spring Core、 Spring AOP、 Spring ORM、 Spring DAO、Spring Context、 Spring Web和 Spring Web MVC[7]。SpringBoot开放源代码、可单独应用于构筑应用程序、可与众多Web框架组合使用、可与桌面应用程序组合使用。

MyBatis数据持久化框架在项目开发过程中负责访问数据库，支持流行的SQL标准化语言，能够帮助开发者避免JDBC代码、手动设置参数和获取数据库结果集[8]。

### 2.4.3 前端框架

首先，前端JSP页面使用HTML语言。利用JavaScript脚本语言和JQuery的validate技术进行前端基本验证，优化用户浏览效果。利用Ajax异步请求技术实现三级联动，最后使用Bootstrap框架进行前端修饰。

## 2.5 本章小结

本章主要针对系统需求和非功能性需求进行分析，介绍开发本系统使用到的前后端主要技术。

# 3 系统概要设计

总体设计是针对系统研究和设计之间的桥梁。其主要目标是根据需求分析，明确模块接口和用户交互界面，设计数据库表和构建目标系统的逻辑模型。

## 3.1 总体业务框架

经过调研、了解学生组织的需要，总结以下主要功能模块，如图3-1所示：

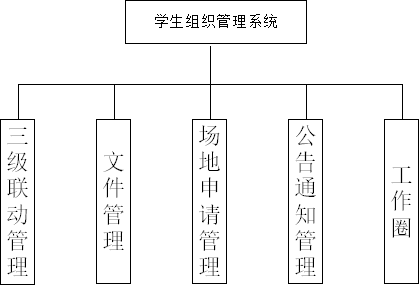


图3-1系统总体功能模块图

## 3.2 浏览器/服务器 架构

Server服务器端负责主要的事务逻辑处理和与数据库进行交互，而Browser浏览器端主要负责与用户之间的交互。

B/S架构的优点在于用户在客户端安装上浏览器便可使用。除此以外，用户不需要手动更新系统，很好地统一了客户端。如图3-2所示：

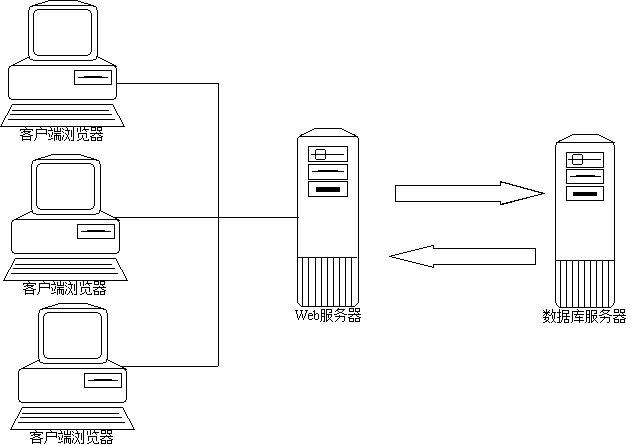


图3-2 B/S架构图

## 3.3 系统总体架构设计

MVC 架构是一种能够将业务逻辑、数据、视图隔离开的设计典范，它能够保证系统具备良好的可扩展性和可维护性。

1、View（视图层）：可以使用HTML、XML编写，结合前段技术JS、JQuery、Ajax、Bootstrap进行修饰优化。也即用户能够之间看到的系统界面。

2、Controller（控制器）：负责接收视图层的请求和参数，处理基本业务逻辑并将数据模型封装送回给视图层，与视图层密切对接。

3、Model（模型层）：对数据库进行CRUD操作，处理逻辑业务，封装结果集数据。

MVC架构如图3-3所示：

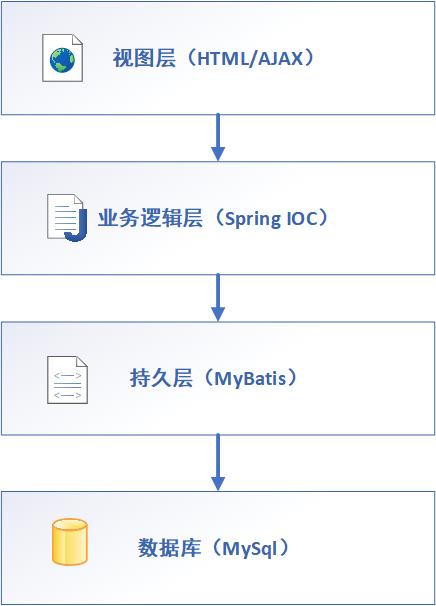


图3-3 MVC架构

## 3.4 功能业务逻辑设计

三级联动管理模块主要负责提供组织、部门以及成员的CRUD管理。组织新增时，教师、管理员有权限创建组织，同时有权新建或修改部门、成员所归属的组织。部门新增时，如学生会准备新增学习部，教师、管理员有权新建部门，同时可以修改成员所归属的部门；学生会等组织纳新时，新成员自行注册、登录，填写个人信息；而普通成员注册成功后只有查询功能。如图3-4所示：

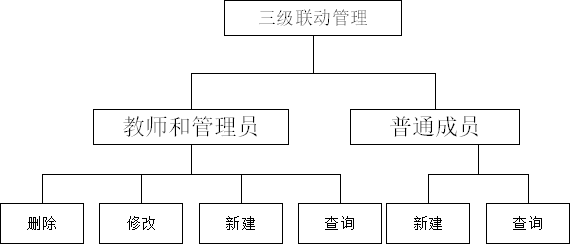


图3-4 三级联动管理功能结构图

文件管理模块主要负责文件的管理。其中包括各组织、各部门文件。所有用户皆可查看文件和上传文件，并且有权删除本人上传的文件。而教师、管理员对系统所有文件拥有删除权限。上传文件时，同时获取上传者所属组织，以供文件查询时通过组织、部门的筛选进行分类查找。如图3-5所示：

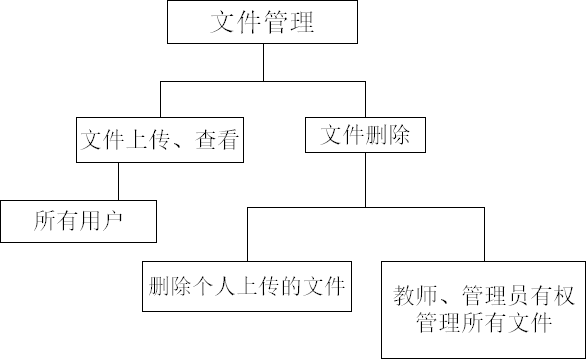


图3-5 文件管理功能结构图

场地申请管理模块主要用于，学生组织开展活动时，为避免与其他组织场地使用冲突，提供申请公共场地功能。以组织部门为单位，确定选择空余地点、确定时间、确定活动内容，提出申请。教师对场地使用申请批准后才能使用，同时教师、管理员可以查询某个场地的使用记录、某组织申请使用过的场地记录。如图3-6所示：

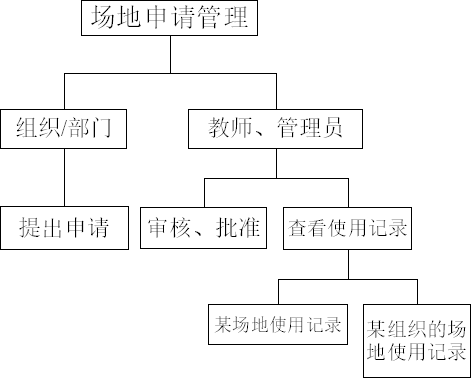
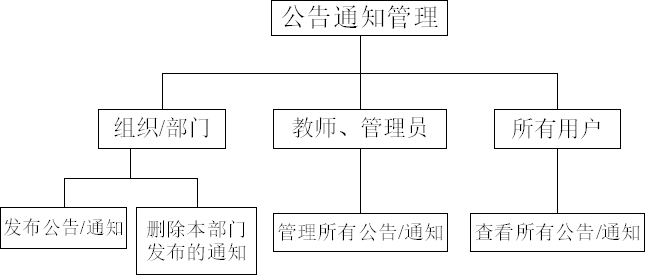


图3-6 场地申请管理功能结构图

公告通知管理中，所有人皆可使用三级联动查询；组织部门管理员可以发布本组织部门的公告通知，并且有权限自行删除；教师、管理员有权对所有公告与通知进行管理。如图3-7所示：

图3-7 公告通知管理功能结构图

工作圈主要用于促进组织部门之间的工作交流。组织/部门举办活动时，所有成员皆可发布活动文案、活动照片，展示在工作圈，并且有权自行删除。同时所有成员可以评论、点赞。如有，教师、管理员有权删除活动展示。如图3-8所示：

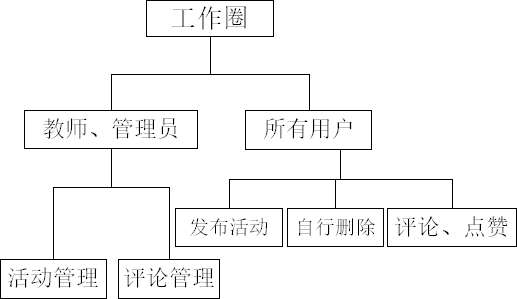


图3-8 工作圈功能结构图

## 3.5 系统数据库设计

数据库的设计是需要依据系统需求，经由软件开发过程当中关键的一步。对给定的系统需求构造最合适的数据库设计。使得本系统的数据存储能够以最优的体式格局。

### 3.5.1 概念结构关系

数据库概念结构关系设计能够将需求分析所得到的毫无结构的数据叙述转换为易于理解且有结构的表达，从而得出合适本系统的数据库概念模型设计[9] ，这是数据库设计过程中一个关键的环节。本文采用E-R图描述概念模型，通过图形所代表的意义和图形间的联系描述本系统中的概念模型。依据需求分析得到的主要E-R图：

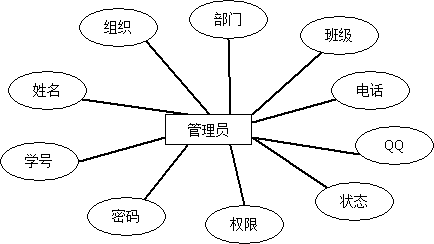


图3-9管理员实体E-R图

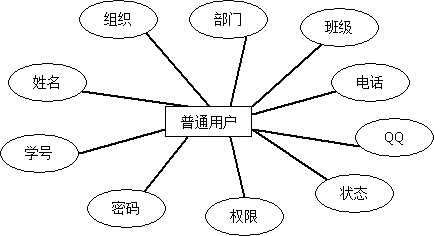


图3-10普通用户实体E-R图

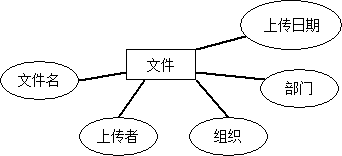


图3-11文件实体E-R图



图3-12组织实体E-R图

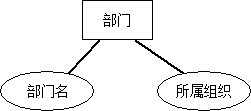


图3-13部门实体E-R图

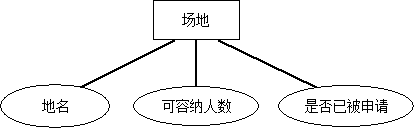


图3-14场地实体E-R图

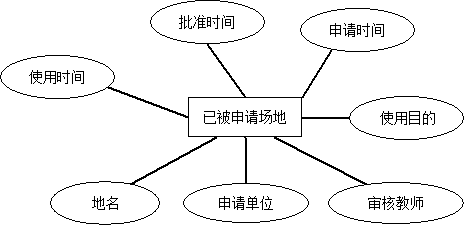


图3-15已被申请场地实体E-R图

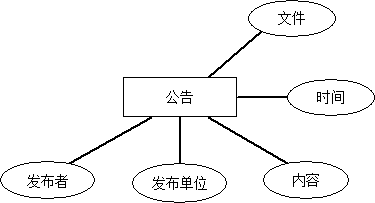


图3-16公告实体E-R图

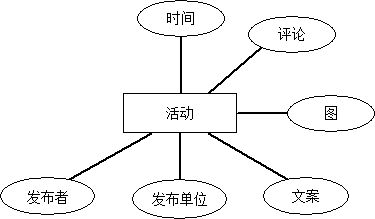


图3-17活动实体E-R图

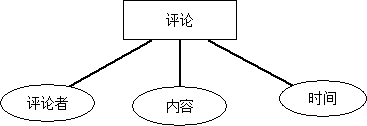


图3-18评论实体E-R图

最后，本系统主要实体关系E-R图如图3-19所示：

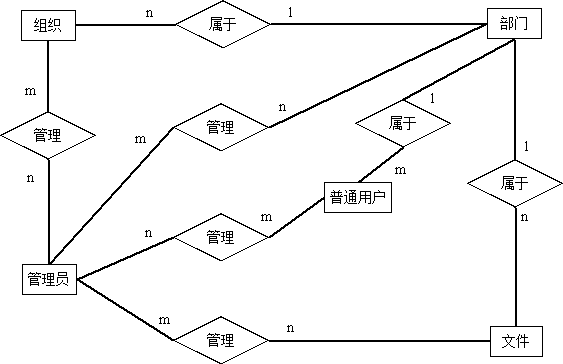


图3-19系统主要实体关系E-R图

### 3.5.2 核心数据表

数据库概念结构设计完成以后，就可以将实体关系图转换为数据表。现主流的数据模型是关系型数据库模型。部分表结构如图：（表3-1至3-10）

表3-1 管理员表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| admin\_id | Int | 10 | \* | id |  |
| admin\_num | Varchar | 20 |  | 学号 |  |
| admin\_name | Varchar | 20 | Not null | 姓名 |  |
| admin\_psw | Varchar | 20 | Not null | 密码 |  |
| admin\_org | Int | 10 |  | 组织 | 外键 |
| admin\_dep | Int | 10 |  | 部门 | 外键 |
| admin\_class | Int | 10 |  | 班级 | 外键 |
| admin\_phone | Varchar | 20 | Not null | 联系电话 |  |
| admin\_QQ | Varchar | 20 |  | QQ |  |
| admin\_state | Bit | 2 | Not null | 状态 |  |
| admin\_right | Int | 2 |  | 权限 |  |

表3-2 普通用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| user\_id | Int | 10 | \* | Id |  |
| user\_num | Varchar | 20 |  | 学号 |  |
| user\_name | Varchar | 20 | Not null | 姓名 |  |
| user\_psw | Varchar | 20 | Not null | 密码 |  |
| user\_org | Int | 10 |  | 组织 | 外键 |
| user\_dep | Int | 10 |  | 所属组织下的部门 | 外键 |
| user\_class | Int | 10 |  | 班级 | 外键 |
| user\_phone | Varchar | 20 | Not null | 联系电话 |  |
| user\_QQ | Varchar | 20 |  | QQ |  |
| user\_state | Tinyint | 2 | Not null | 状态 |  |
| user\_right | Tinyint | 2 |  | 权限 |  |

表3-3 文件表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| file\_id | Int | 10 | \* | Id |  |
| file\_name | Varchar | 255 |  | 文件原始名 |  |
| file\_uploader | Int | 10 |  | 文件上传者 | 外键 |
| file\_dep | Int | 10 |  | 部门 | 外键 |
| file\_date | Datetime | 6 |  | 上传日期 |  |

表3-4 组织表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| org\_id | Int | 10 | \* | Id |  |
| org\_name | Varchar | 50 |  | 组织名 |  |

表3-5 部门表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| dep\_id | Int | 10 | \* | Id |  |
| dep\_name | Varchar | 50 |  | 部门名 |  |
| dep\_org | Int | 4 |  | 部门所属组织 | 外键 |

表3-6 场地表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| place\_id | Int | 10 | \* | id |  |
| place\_name | Varchar | 80 |  | 场地名 |  |
| place\_capacity | Int | 6 |  | 可容纳人数 |  |
| place\_Used | tinyint | 2 |  | 是否已使用 |  |

表3-7 已被申请场地表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| usedP\_id | Int | 10 | \* | id |  |
| usedP\_D | int | 10 |  | 申请单位 | 外键 |
| usedP\_teacher | Int | 10 |  | 审核管理员 | 外键 |
| usedP\_For | Varchar | 255 |  | 使用目的 |  |
| usedP\_applyT | datetime | 6 |  | 申请时间 |  |
| usedP\_allowT | datetime | 6 |  | 批准时间 |  |
| usedP\_startT | datetime | 6 |  | 开始使用时间 |  |
| usedP\_endT | datetime | 6 |  | 结束使用时间 |  |
| usedP\_place | Int | 10 |  | 场地 | 外键 |

表3-8 公告表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| notice\_id | Int | 10 | \* | id |  |
| notice\_person | Int | 10 |  | 发布者 | 外键 |
| notice\_dep | Int | 10 |  | 发布单位 | 外键 |
| notice\_content | Varchar | 255 |  | 发布内容 |  |
| notice\_time | datetime | 6 |  | 发布时间 |  |
| notice\_file | Varchar | 255 |  | 文件链接 |  |

表3-9 活动表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| active\_id | Int | 10 | \* | id |  |
| active\_person | Int | 10 |  | 发布者 | 外键 |
| active\_dep | Int | 10 |  | 发布单位 | 外键 |
| active\_content | Varchar | 255 |  | 文案内容 |  |
| active\_picture | Varchar | 255 |  | 配图 |  |
| active\_conment | Int | 10 |  | 评论 | 外键 |
| active\_time | Datetime | 20 |  | 发布时间 |  |

表3-10 评论表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| comment\_id | Int | 10 | \* | id |  |
| comment\_activeId | Int | 10 |  | 所评论的活动 | 外键 |
| comment\_person | Int | 10 |  | 评论者 | 外键 |
| comment\_time | Datetime | 20 |  | 评论时间 |  |

表3-11班级表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 主键 | 说明 | 其他 |
| class\_id | Int | 10 | \* | id |  |
| class\_name | Varchar | 50 |  | 班级 |  |

## 3.6 本章小结

本章主要分析系统的总体架构、功能业务逻辑和数据库的设计。

# 4 系统功能模块详细设计与实现

结合前面章节的分析，本章更加深入地阐述功能模块的详细设计与具体实现，主要介绍个别模块。

根据数据库表设计，我们将对象转变成实体类，表属性转换成实体属性。本系统的主要实体类如图4-1：

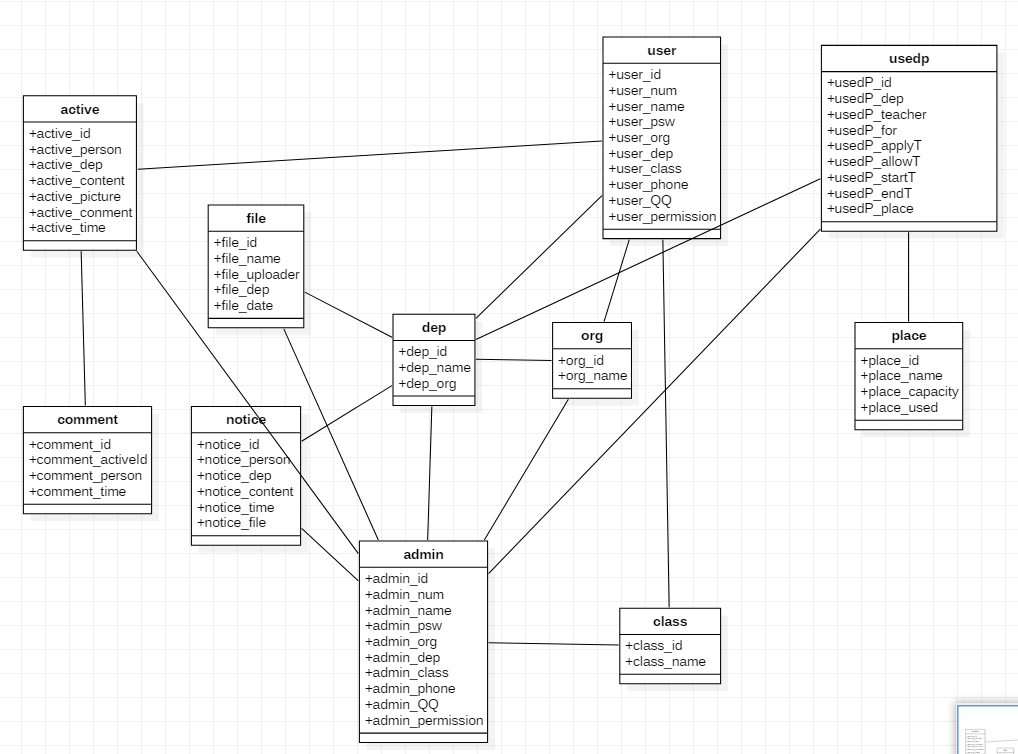


图4-1 主要实体类

根据本系统的架构设计，用户在JSP界面点击发出请求时。页面将必要的数据参数传送至Controller。Controller响应、处理事务，使用数据调用Service，Service会做异常处理，同时调用Mapper。对数据库的基本操作由Mapper封装。Mapper对数据库CRUD操作完毕以后返回有效数据给Service。Service返回给Controller。Controller提供转发页面或者重定向页面，并返回有效数据至JSP页面。具体层级调用顺序图如图4-2所示：

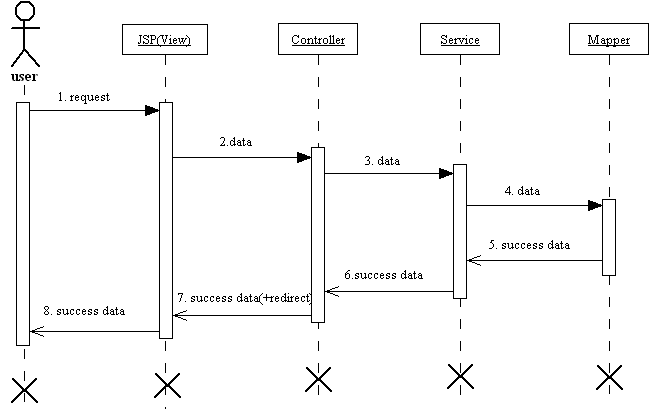


图4-2层级调用顺序图

## 4.1 页面登录注册功能详细设计

### 4.1.1 用户注册功能

游客想要注册成为用户：首先输入基本信息，其次判断该游客号码是否已被注册。若号码已经被注册，则不必继续注册，可以直接跳转登录界面。若号码未被注册，则可以注册。注册功能的基本流程图如图4-3:

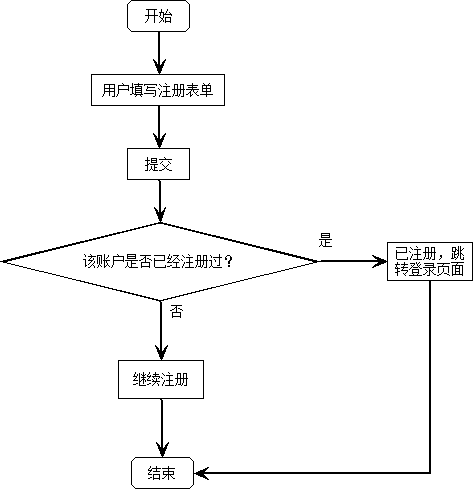


图4-3 注册功能流程图

### 4.1.2 用户、管理员登录功能

首先，用户输入账号、密码并选择角色，点击提交，后台Controller需要先判断用户是否存在于数据库中。若用户不存在则提示。若用户存在，则判别密码。若密码正确则登录成功。若密码不正确则提示有误，请用户重新输入。登录功能的基本流程图如图4-4所示：

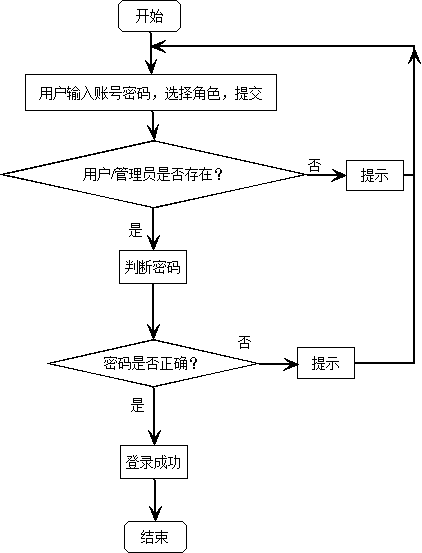


图4-4登录功能流程图

## 4.2功能模块详细设计

三级联动管理：管理员进入管理页面以后，起初显示全部的组织信息：OrgMapper提供findAll()查找所有组织。对组织进行添加、删除和修改：OrgMapper提供addOrg()、delOrg()和updateOrg()。

当点击管理其部门时，进入部门管理页面：DepMapper提供findByOrg()方法根据orgId查找部门。对部门进行添加、删除和修改：DepMapper提供addDep()、delDep()和updateDep()。

最后则是点击管理成员，进入用户管理界面，UserMapper提供findByDep()查找本部门的全部成员。对部门下的成员进行管理：UserMapper提供addUser()、delUser()和updateUser()。

此功能涉及到的主要类图如图4-5所示：

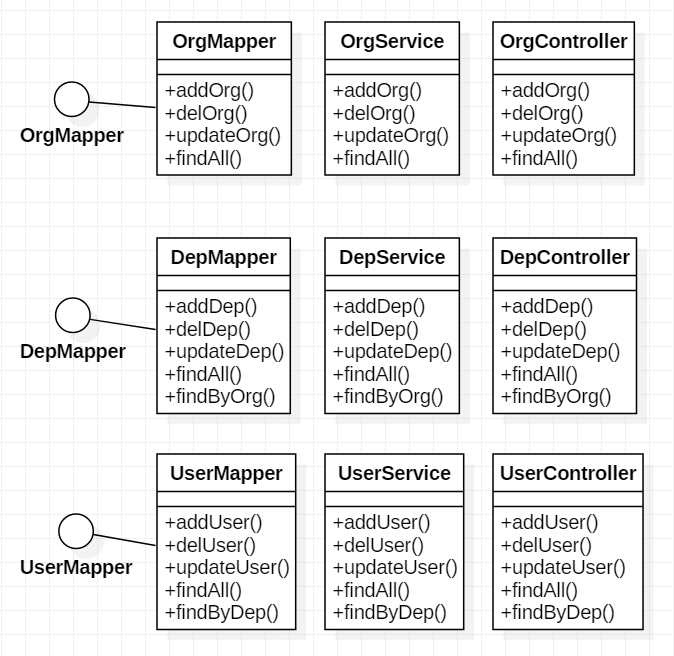


图4-5三级联动管理模块主要类图

文件管理界面可以通过选择不同组织不同部门，从而筛选文件：DepMapper提供findByOrg()方法、FileMapper提供findByDep()。用户进入文件管理页面，可以下载文件:FileMapper提供findAll()。管理员可以上传、删除、下载文件:FileMapper提供addFile()和delFil。此功能涉及到的主要类图如图4-6所示：

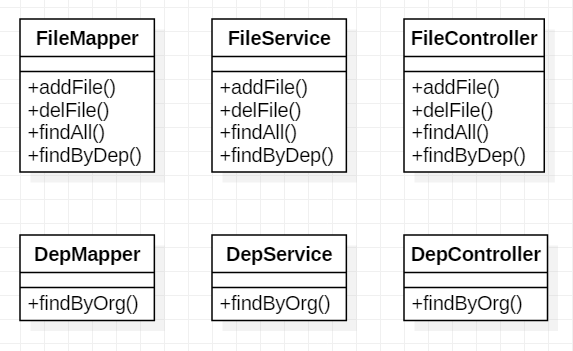


图4-6文件管理模块主要类图

场地申请管理：组织/部门提出场地申请：PlaceMapper提供findAll()显示所有场地。UsedpMapper提供addUsedp()。同时要更新place表的place\_used字段标志该场地已被使用，以免其他部门继续提出申请：PlaceMapper提供updatePlace()。

管理员对场地申请进行审核批准：UsedpMapper提供findAll()查找updateUsedp()方法批准。

管理员查看使用记录：UsedpMapper提供findByPlace()查看某场地使用记录。UsedpMapper提供findByDep()查看某组织的场地申请记录。此功能涉及到的主要类图如图4-7所示：

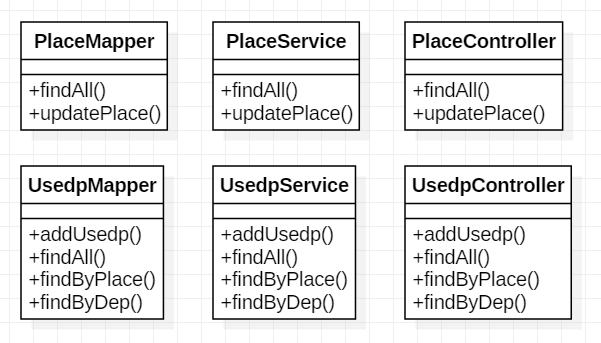


图4-7场地申请模块主要类图

公告通知管理：用户查看公告时，可以根据组织和部门进行筛选：NoticMapper提供findAll()和findByDep()，DepMapper提供findByOrg()。管理员对公告的管理：NoticeMapper提供addNotice()、delNotice()、updateNotice()。此功能涉及到的主要类图如图4-8所示：

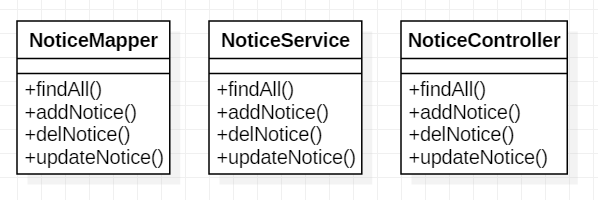


图4-8公告通知管理模块主要类图

工作圈：用户发布活动、同时有权删除自己发布的活动：ActiveMapper提供addActive()、delActive()。用户对某条活动进行评论，并且有权删除：CommentMapper提供addComment()、delComment()。

管理员对于活动的管理：ActiveMapper提供findAll()查找活动，delActive()删除活动。管理员对于评论的管理：CommentMapper提供findByActive()查找到某活动的评论，delComment()删除评论。

此功能涉及到的主要类图如图4-9所示：

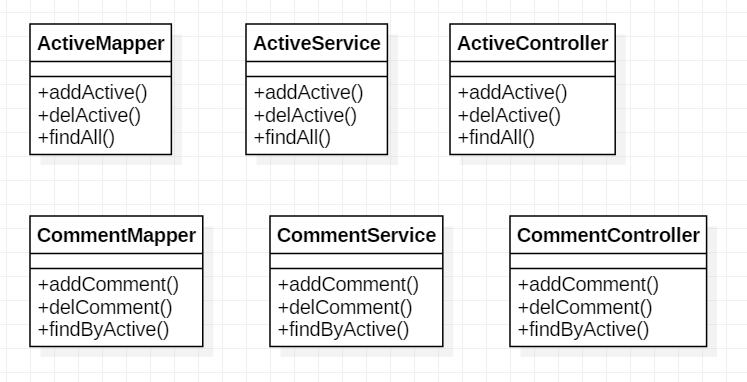


图4-9 工作圈模块主要类图

## 4.3 系统界面的设计与实现

本小节将展示系统前端部分界面截图如图4-10至秃4-18：

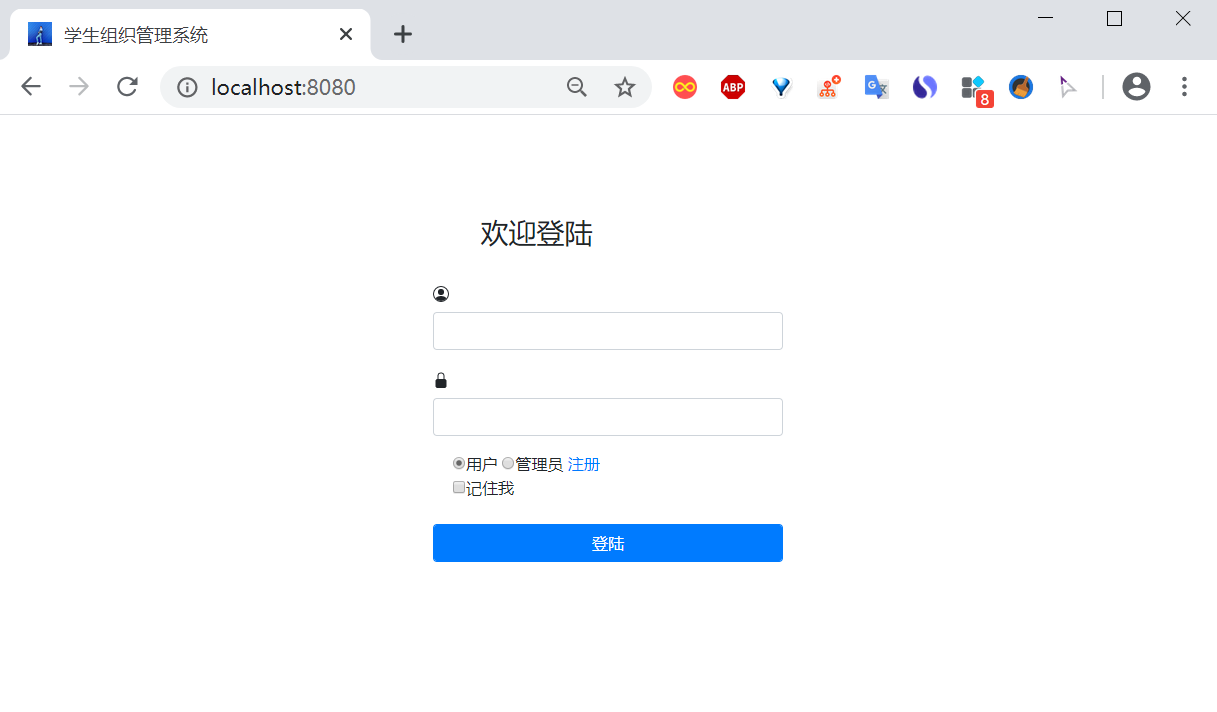


图4-10 用户管理员登录界面

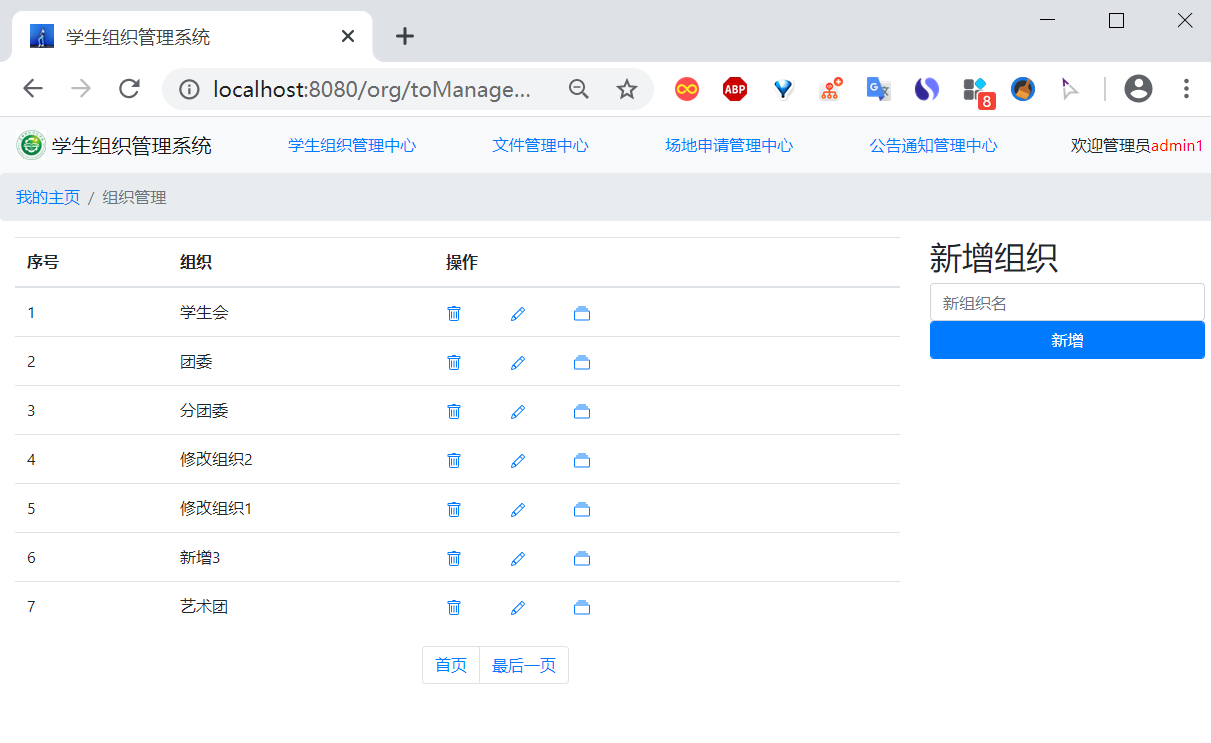


图4-11 组织管理界面

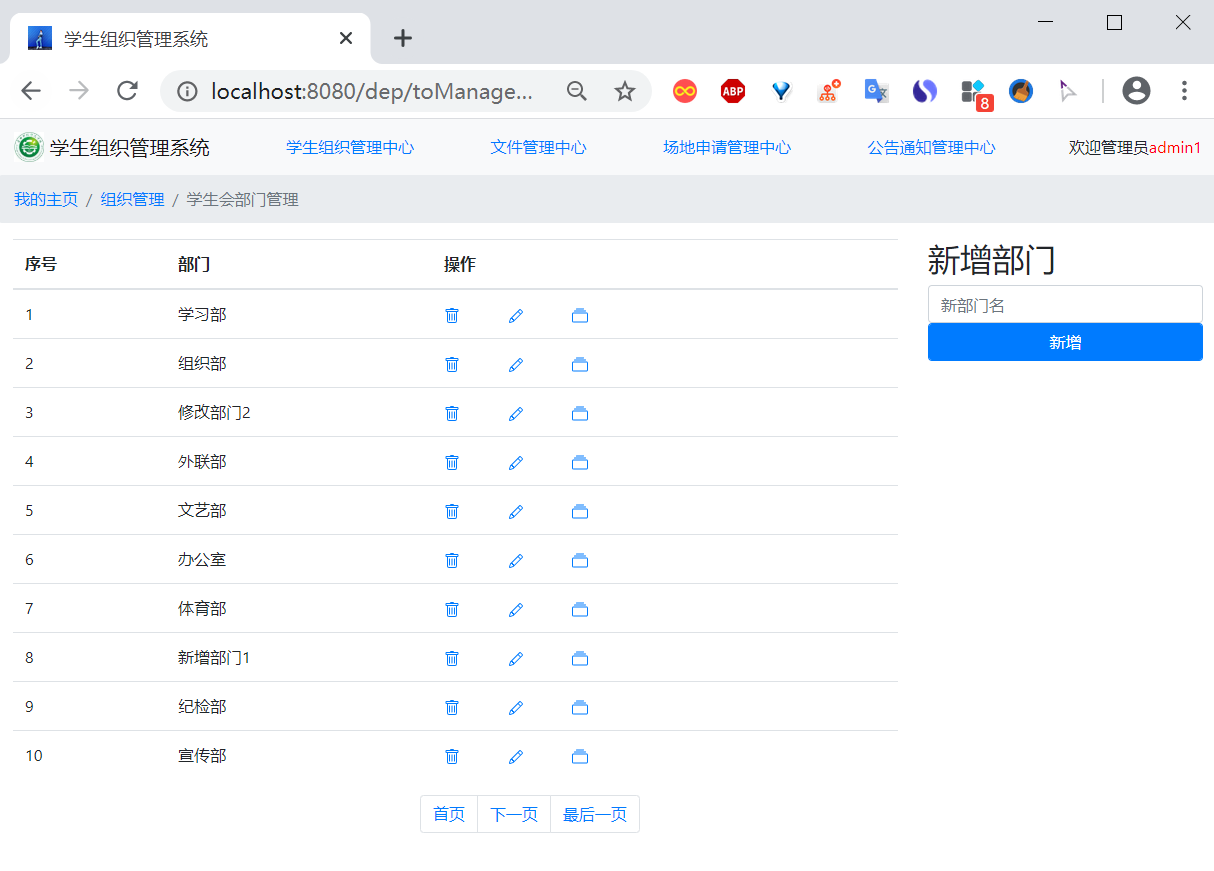


图4-12 部门管理界面

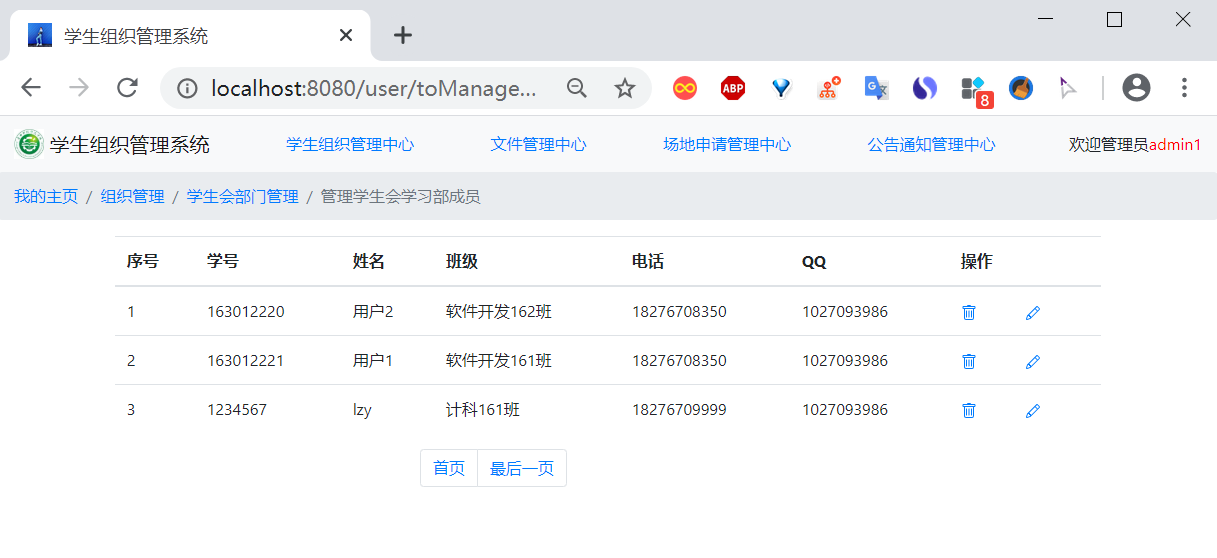


图4-13 用户管理界面

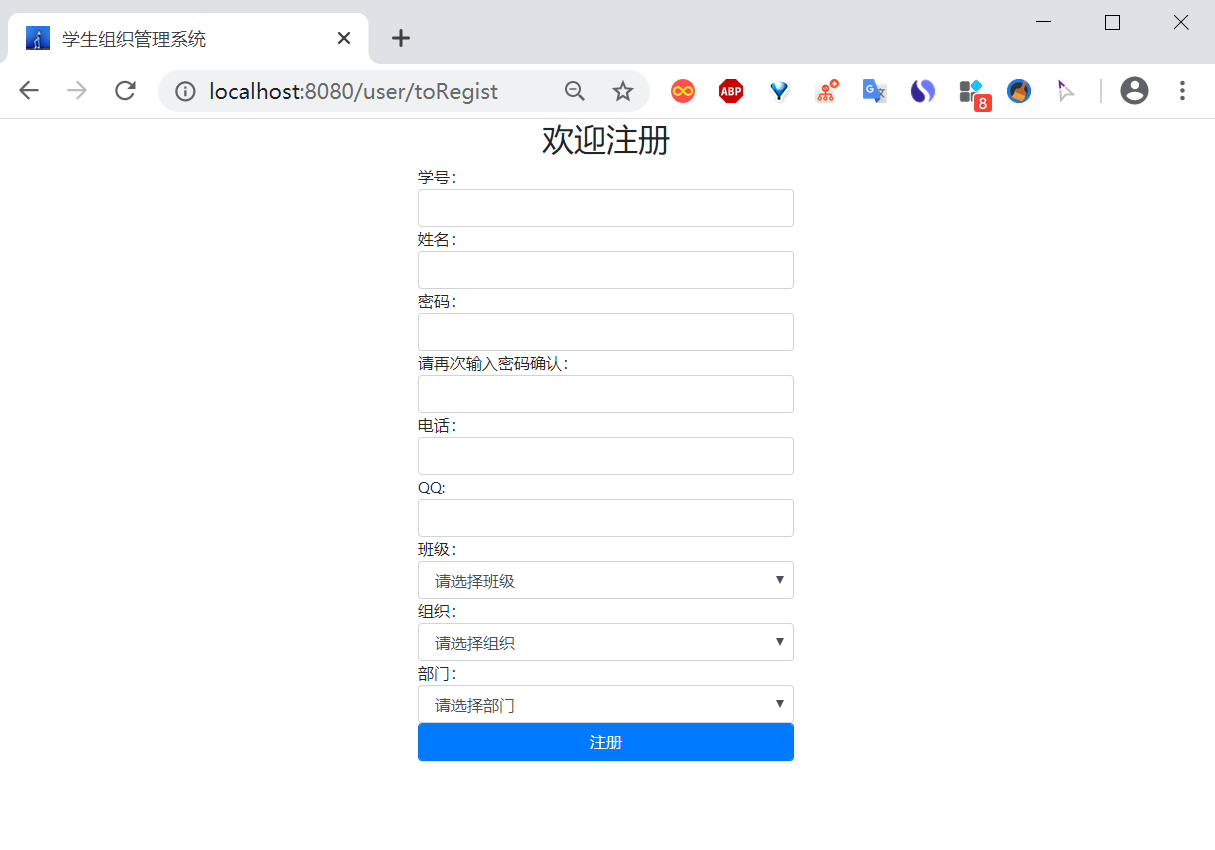


图4-14 用户注册界面

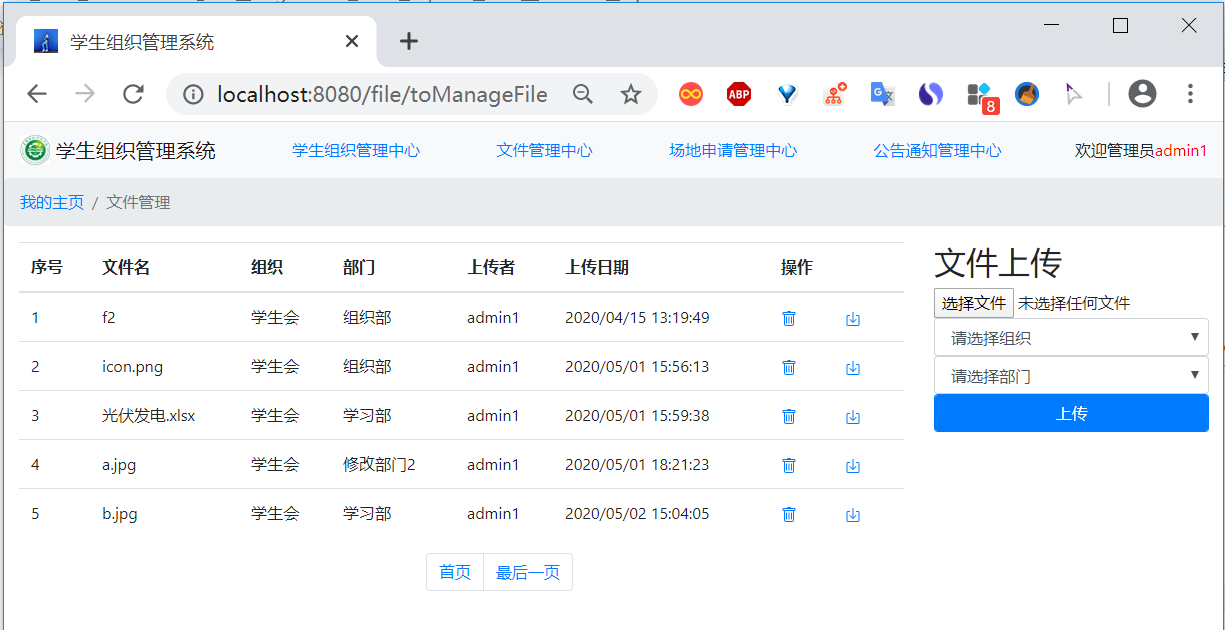


图4-15 文件管理界面

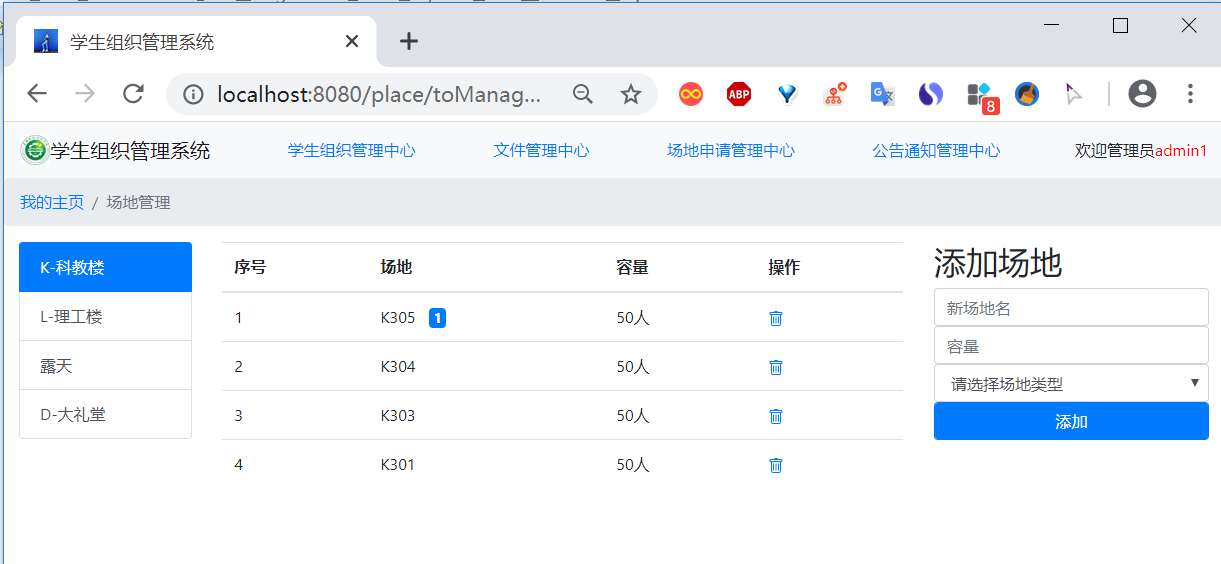


图4-16 场地管理界面

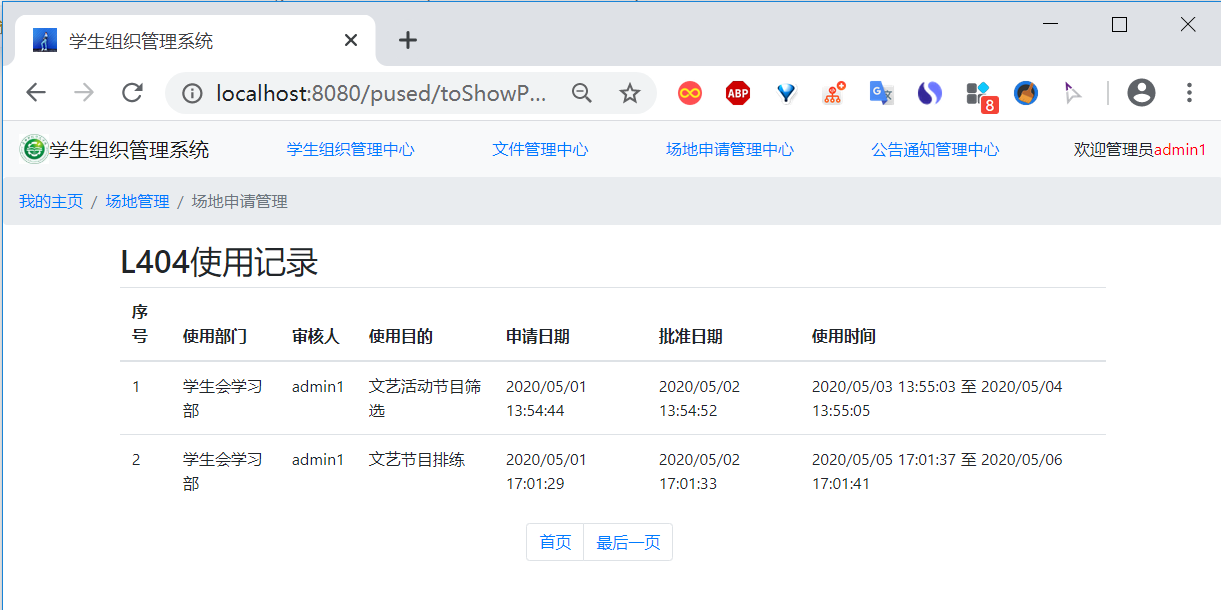


图4-17 场地使用记录界面

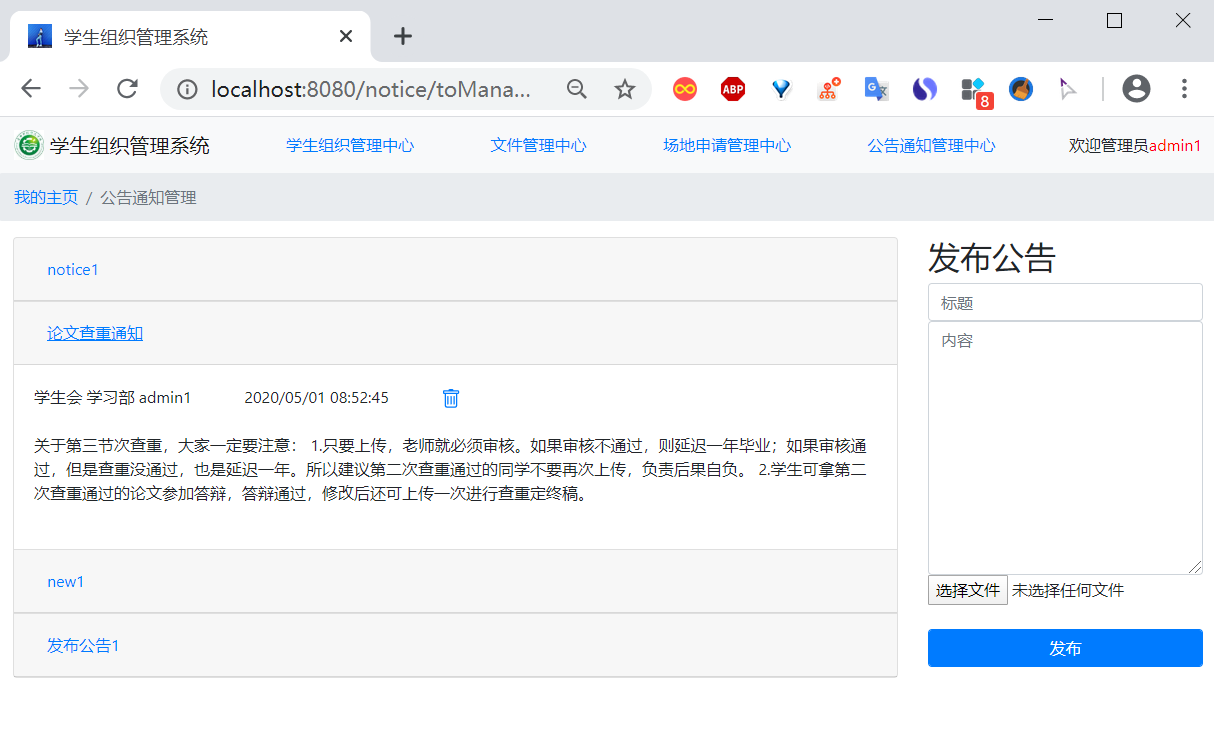


图4-18 公告通知管理界面

## 4.4 本章小结

本章对系统各个主要功能进行了详细的设计描述。给出系统开发设计中的具体层级调用顺序并且提供系统主要类图和主要方法。

# 5 系统功能模块测试

本章将针对系统功能模块展开系统功能测试。我们需要测试验证系统是否能够正常使用以及是否能够按照预期设计正常运行，以达到系统可交付使用的目的。

## 5.1 系统测试环境

Java 版本：1.8.0

服务器：Apache Tomcat v8.0

运行环境JRE：jdk1.8.0\_152

浏览器：Google Chrome 80.0.3987.162（正式版本）（32 位）

操作系统：Windows10

数据库：mysql8.0.15

## 5.2 功能测试内容

依据需求分析，对系统开展功能性测试。目的在于检查系统存在的错误并加以修正解决。本小结主要的功能测试有：

* 基本登录注册功能的正常使用。
* 三级联动管理功能：管理员登录之后是不是可以正常使用三级联动管理。
* 文件管理功能：管理员登录之后是不是能够对文件进行管理。用户是否能够正常下载文件。
* 场地申请管理功能：用户登录以后能否正常使用场地申请，管理员登录后能否进行管理操作。
* 公告通知管理功能：管理员登陆以后能否对公告进行正常的管理。
* 工作圈功能：用户是否能够正常发布活动，管理员是否能够正常管理活动与评论。

## 5.3 主要测试用例

针对该系统需求，本小节将给出主要的测试用例：

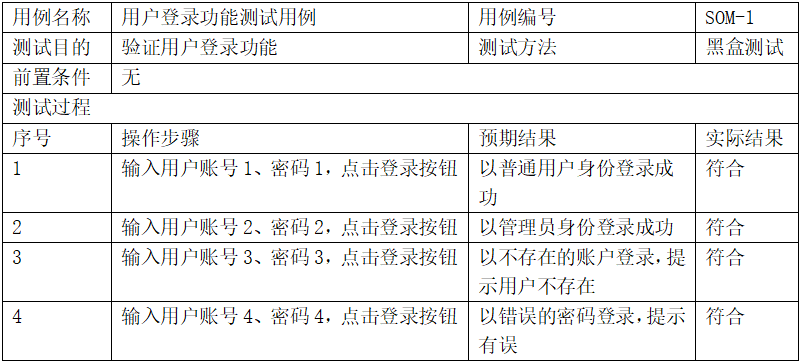


图5-1 用户登录功能测试用例

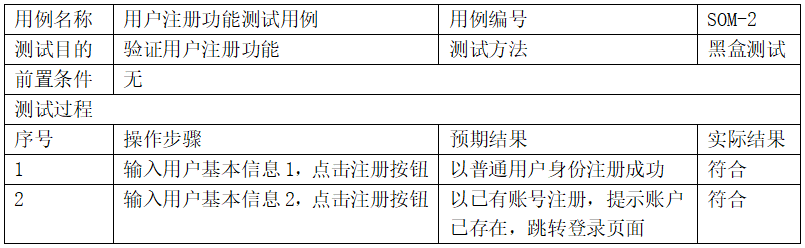


图5-2 用户注册功能测试用例

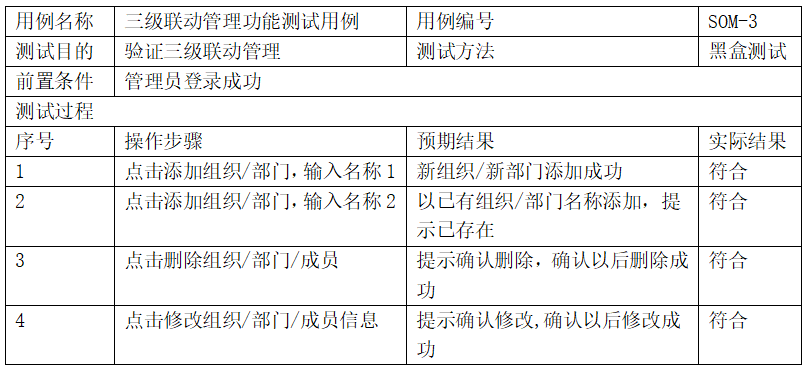


图5-3 三级联动管理功能测试用例

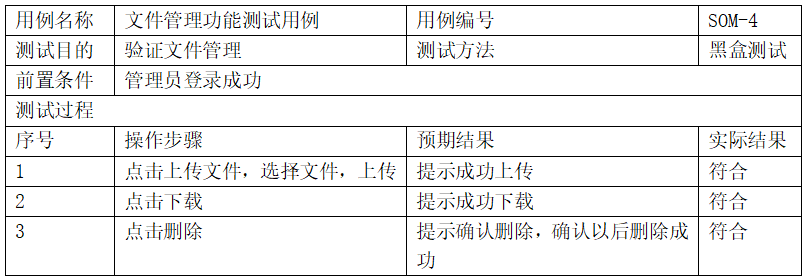


图5-4 文件管理功能测试用例

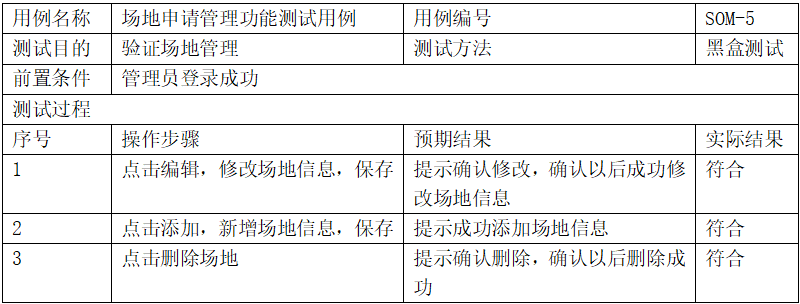


图5-5 场地申请管理功能测试用例

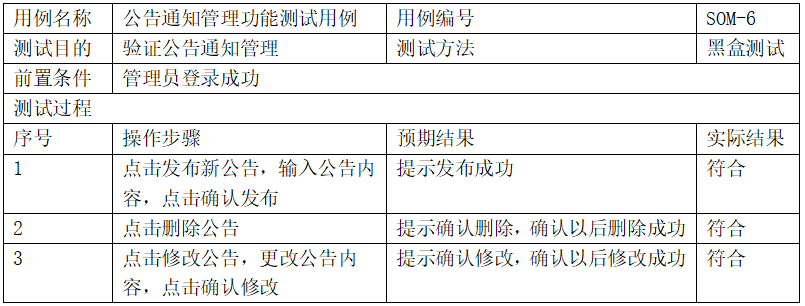


图5-6 公告通知管理功能测试用例

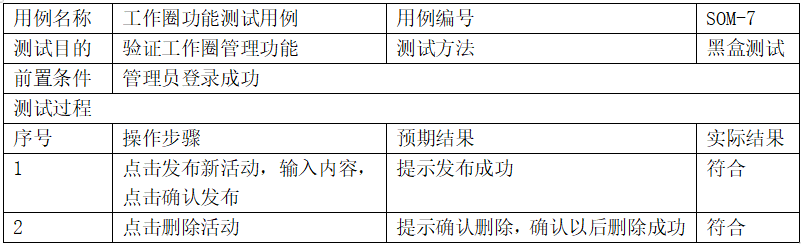


图5-7 工作圈功能测试用例

## 5.4 本章小结

本章对系统的测试给出总体描述。首先介绍使用的测试环境，其次介绍需要测试的内容，最终给出主要的测试用例。通过对系统的测试并加以修正，保证本系统满足学生组织管理的基本需求。

# 6 总结与展望

## 6.1全文总结

本文主要通过分析高校学生组织管理中存在的问题来构建系统化的学生组织管理系统，针对工作冲突、分散孤立、管理效率低、管理混乱等问题，提出基于web的高校学生组织设计的管理系统，有效地提高了教师们在学生组织管理上的专业性，同时也有助于学生组织的发展。本文主要完成以下几项工作：

1. 开展学生组织管理调研，通过比对当前的研究现况完成系统需求分析，分析本管理系统对于高校学生会、团委会等学生组织的用处。
2. 设计并实现了基于Web的高校组织管理系统，该系统提供三级联动管理、文件管理、场地申请管理、公告通知管理等功能，通过对后台接口的调用，实现前后端分离。最后通过Web端展示各种基础功能，方便用户掌握实际的学生组织情况。

## 6.2系统存在问题和未来展望

由于时间受限、本人能力有限和实战经验限制，在该系统实现过程中依然存在诸多工作方法尚待改进，同时本系统也存在需要改良优化的问题，主要有下述几个方面：

1. 软件需求分析是软件工程中的开头第一大重要环节，需求分析如果再详细明确一些，要实现的功能自然也能够更加明了。
2. 功能模块详细设计的逻辑事物实现有待改进，在编写代码工作上，需要更加严谨。代码规范，逻辑明确，才能够利于系统的维护。
3. 在测试工作上，测试环境单一，不同的使用者所使用的系统环境、浏览器琳琅满目，应该在更多不同的环境下做测试。
4. 便利智能手机的普及化，使得普遍的用户在日常生活中更趋向于使用智能手机，期望在未来能够为该系统编写基于Android系统和IOS系统客户端。

# 参考文献

[1] 张明东,戴丹丹.Java EE下学生信息管理系统的设计与实现[J].才智,2018(08):164.

[2] 初庆东,张敏霞,崔卫韬.一种新型高校学生信息管理系统的设计与实现[J].信息技术,2019,43(06):69-71+76.

[3] 杨聪颖. 福建广播电视大学学生信息综合管理系统的设计与实现[D].天津大学,2018.

[4] 韩彪. 贵州师范学院团员信息管理系统的研究与分析[D].云南大学,2016.

[5] 汤金,武钊,曾卿宇.基于Java的学生管理系统的设计与实现[J].计算机产品与流通,2019(06):240.

[6] 王明贵.分权与边界:基于对美国高校中层管理考察的探讨——以美国罗格斯大学为例[J].教育文化论坛,2019,11(05):86-90.

[7] 廖义奎编著.Java Web开发之Struts编程基础与实例精讲[M].中国电力出版社.2006年1月.第372页

[8] 王丽,宗镇欣.基于SSM框架的学生信息管理系统[J].电脑知识与技术,2019,15(28):95-96.

[9] Hong ZHANG. Proceedings of the 2019 3rd International Conference on Education, Management Science and Economics (ICEMSE 2019)[C]. Atlantis Press, Proceedings of the 2019 3rd International Conference on Education, Management Science and Economics (ICEMSE 2019),2019.

[10] 孙祥玺. 某中学“家校通”信息管理系统的分析与设计[D].北京邮电大学,2017.

[11] 王微漪. 基于Web的高职院校学生信息管理系统的设计与实现[D].北京工业大学,2017.

[12] Stanley B.Lippman / Josee Lajoie / Barbara E.Moo ．C++ Primer[M]：人民邮电出版社，2010

[13] Venu Challagolla,Avon,CT(US);Mark R.Finn,Bristol,CT(US);Gai Sai Kiran Paladugu,Manchester,CT(US),Code Development Management System[P].US2019347093.Nov.14,2019.

[14] 刘丽华,薛玉倩.基于spring MVC的学生信息管理系统[J].通讯世界,2019,26(10):173-174.

[15] 王浩,邢稚.基于NSP.NET的学生信息管理系统[J].中国新通信,2019,21(01):119.

[16] 张立新.学生信息管理系统的设计与实现[J].中国多媒体与网络教学学报(中旬刊),2019(02):37-38.

[17] 李梦槐.基于SSH框架的学生信息管理系统的设计与实现[J].江苏科技信息,2018,35(16):52-54.

[18] 彭毅超. 基于B/S模式的学生信息管理系统设计与实现[D].湖南大学,2017.

# 致 谢

在毕业论文完成之际，我想对我的老师、同学、朋友以及家人一直以来的支持关系和帮助表示衷心的感谢。

感谢李志明老师，我的论文指导老师，在完成毕业设计的过程中耐心的指导和热情的帮助。任何时候，当我提出问题，老师都会在百忙中放下手中的工作给予及时的指导，老师对于工作的负责态度和学识的渊博鼓舞了我。李志明老师在论文写作措辞上要求严格，在论文编写过程指导中开拓了我的思维模式。感谢程州老师，在毕业设计上给予我的建议和帮助。程州老师对于我提出的问题都回答得相当细心，并且在代码技术上非常得手。正是函授老师百忙之中的多次审阅，对本文细节的修饰和全文的撰写给予了很多宝贵意见，才使得本文成型。

真诚地感激数学与计算机科学学院每一位老师，你们的教诲为本文提供了理论基础，尤其在思想方法上给了我很多新的启发。思念时光短暂，但师生情很长，这些老师教会的知识、道理将伴随我走向未来。

感恩亲爱的同学们、朋友们，快乐的时光离不开你们的相伴！

感谢我的家人们，你们伟大的父爱母爱让我感受到了家庭的温暖。谢谢你们一直以来对我的理解和支持！

最后，由衷地感谢百忙之中抽空审阅本论文的专家老师。由于本人写作水平有限，本文难免有僻陋，恳请老师与同学指教。

# 作者简介

姓名：李峥滢 女 1998年9月 汉族

数学与计算机科学学院计算机科学与技术专业2020届本科生

E－mail:18276708350@163.com

附录

无

# 声 明

本人郑重声明：所成交的毕业论文（设计），题目：《基于Web的学生组织管理系统设计与实现》是本人在广西科技师范学院学习期间，在指导教师指导下进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，其内容真实可靠，如存在抄袭、剽窃、伪造现象，本人愿承担全部责任。

同时，本人完全了解并愿意遵守广西科技师范学院有关保存、使用毕业论文（设计）的规定，其中包括：

1.学院有权保管并向有关部门递交毕业论文（设计）的原件与复印件。

2.学院可以采用影印、缩印或其他复制方式保存毕业论文（设计）。

3.学院可以以学术交流为目的，赠送和交换毕业论文（设计）。

4.学院可以允许毕的业论文（设计）被查阅或借阅。

5.学院可以按著作权法的规定公布毕业论文（设计）的全部内容或部分内容（保密毕业论文（设计）在解密后遵守此规定）。

6.除非另有科研合同或其他法律文书制约，本论文的科研成果属于广西科技师范学院。

特此声明！

声明人签名：李峥滢

二〇二〇年五月二十五日