**目 录**

[**1 绪 论 1**](#_Toc13916)

[1.1 研究的背景及意义 1](#_Toc31967)

[1.1.1 选题的背景 1](#_Toc26898)

[1.1.2 国内外研究现状 1](#_Toc19829)

[1.1.3 研究的意义 1](#_Toc11465)

[1.2 系统目标 2](#_Toc4867)

[**2 需求分析 3**](#_Toc5800)

[2.1 业务需求 3](#_Toc24558)

[2.1.1主要业务流程 3](#_Toc25071)

[2.2 功能需求 10](#_Toc29472)

[2.2.1 角色分析 10](#_Toc10211)

[2.2.2 业务功能 10](#_Toc1732)

[2.3 非功能需求 43](#_Toc4119)

[2.3.1环境需求 43](#_Toc27114)

[2.3.2 性能需求 43](#_Toc1018)

[2.3.3 安全需求 44](#_Toc25698)

[**3 总体设计 45**](#_Toc2000)

[3.1系统设计的原则](#_Toc7753)

[3.2 系统体系结构设计](#_Toc11394)

[3.3 系统功能结构设计](#_Toc20764)

[**4 数据库设计 48**](#_Toc26861)

[4.1 概念结构设计 48](#_Toc30289)

[4.1.1 设计思路 48](#_Toc24207)

[4.1.2 E-R图 49](#_Toc1987)

[4.2 逻辑结构设计 55](#_Toc27853)

[4.2.1 设计思路 55](#_Toc18548)

[4.2.2 逻辑模型 55](#_Toc20253)

[4.3 物理结构设计 56](#_Toc118)

[4.3.1 存取方式 56](#_Toc9695)

[4.3.2 存储结构 56](#_Toc11510)

[**5 界面设计 63**](#_Toc1710)

[5.1 界面关系图或工作流图 63](#_Toc28092)

[5.2 界面设计成果 63](#_Toc22628)

[5.2.1 主界面 63](#_Toc930)

[5.2.2 子界面 65](#_Toc8244)

[**6 详细设计 70**](#_Toc4048)

[6.1 系统主要功能模块介绍 70](#_Toc20599)

[6.2用户管理 71](#_Toc444)

[6.2.1 功能结构设计 71](#_Toc28464)

[6.2.2 类图设计 71](#_Toc21052)

[6.2.3 顺序图设计 73](#_Toc22859)

[6.2.4 核心处理流程设计 74](#_Toc23606)

[6.3项目管理 74](#_Toc23018)

[6.3.1功能结构设计 74](#_Toc7032)

[6.3.2类图设计 75](#_Toc27775)

[6.3.3顺序图设计 77](#_Toc13447)

[6.3.4核心处理流程设计 77](#_Toc30345)

[6.4阶段管理 78](#_Toc16352)

[6.4.1功能结构设计 78](#_Toc23731)

[6.4.2类图设计 79](#_Toc26336)

[6.4.3顺序图设计 81](#_Toc1244)

[6.4.4核心处理流程设计 81](#_Toc6675)

[6.5提交材料管理 82](#_Toc24281)

[6.5.1功能结构设计 82](#_Toc19826)

[6.5.2.类图设计 83](#_Toc11355)

[6.5.3顺序图设计 85](#_Toc11063)

[6.5.4核心处理流程设计 85](#_Toc18638)

[6.6考勤管理 86](#_Toc6836)

[6.6.1功能结构设计 86](#_Toc31089)

[6.6.2类图设计 87](#_Toc8856)

[6.6.3顺序图设计 89](#_Toc20220)

[6.6.4核心处理流程设计 89](#_Toc10870)

[6.7成绩管理 90](#_Toc11651)

[6.7.1功能结构设计 90](#_Toc32156)

[6.7.2类图设计 91](#_Toc29825)

[6.7.3顺序图设计 93](#_Toc16755)

[6.7.4核心处理流程设计 93](#_Toc20527)

[**7 编码 95**](#_Toc25593)

[7.1 代码实现与核心算法](#_Toc18062)

[7.2 代码优化分析](#_Toc27429)

[**8 测试 101**](#_Toc26065)

[8.1 测试方案设计](#_Toc5157)

[8.1.1 测试策略 101](#_Toc26380)

[8.1.2 测试进度安排 101](#_Toc27101)

[8.1.3 测试资源 101](#_Toc28492)

[8.1.4 关键测试点 102](#_Toc29711)

[8.2 测试用例构建 102](#_Toc10886)

[8.2.1 测试用例编写约定 102](#_Toc27922)

[8.2.2 测试用例设计 103](#_Toc5895)

[8.2.3 关键测试用例 103](#_Toc6668)

[8.2.4 测试用例维护 105](#_Toc19779)

[**9 总结与展望 106**](#_Toc505)

[9.1 设计工作总结](#_Toc12085)

[9.2 未来工作展望](#_Toc25108)

[**谢 辞 107**](#_Toc946)

[**参考文献 108**](#_Toc29271)

[**附录A 外文翻译—原文部分 109**](#_Toc25946)

[**附录B 外文翻译—译文部分 115**](#_Toc25390)

[**附录C 软件使用说明书 120**](#_Toc27148)

[**附录D 主要源代码 125**](#_Toc1428)

1 绪 论

## 1.1 研究的背景及意义

### 1.1.1选题的背景

随着互联网的不断发展，越来越多的学生、老师、企业、学校之间希望通过一个平台解决校企之间合作的实习和培训问题，校企合作教学管理水平的高低在一定程度上反映了一个学校的人才培养的质量和水平。对于本校来说校企合作教学管理工作涉及专业课程较多，所涉及的专业技术领域较广、数量也比较多。校企合作教学管理是一项琐碎、复杂而又十分细致的工作。传统的校企合作，学校每年需要花费大量的人力物力去为学生到处挖掘实训、实习企业，寻找企业工程师到校授课，而且查找的途径由于等约束往往比较局限，使得校企合作的管理成本不断提高。

此系统作用在校企老师，学生身上，他们需要注册并登录账号，潜在用户是系统管理员，系统管理员负责管理此系统下发给企业教师的实训任务、实训主题、实训前材料、班级信息以及各种用户的消息，企业老师则负责对学生实习和培训的项目进行管理，包括学生组建的团队、实训的项目、项目里包含的阶段、与阶段相关的参考资料、学生实训时上传提交的材料、学生的考勤信息、实训后的项目成绩。不过此系统也有不足之处，例如考勤系统只需签到并没有特定的位置用来支持签到，还有系统审核员的审核有些文案不一定能够提交成功。

### 1.1.2 国内外研究现状

在现如今这个高科技盛行的时代，仅仅通过传统的管理方式进行学校、企业之间的合作实习、培训管理，已远远不能够达到我们所期盼的结果，我们渴望高效的完成签到管理、学生教师管理、材料文档管理、学生实训的成绩管理。如今的校企合作教学管理工作这一块需要做一个系统去管理，使得校企任课老师与学校学生信息情况进行充分的了解。对学生消息的通知，对学生布置任务，同时能够在网上平台查看并批阅学生的任务，由此做到管理学生，代替之前的人工型手动管理，能让任课老师更加轻松，同时也能让学校了解到自己本校的学生的动态以及学生在校企合作班学习的基本信息。

此系统操作简单，功能模块的实用性强，但是对职业教育的培养重点不同而使得人才培养模式、课程体系和管理模式上具有差别，而且国外的软件系统很难切合到国内的业务需求，使得在应用上存在很大的问题。

### 1.1.3 研究的意义

当时代今，仅通过聊天软件彼此之间传送文件、查看项目进度、进行团队管理、阶段管理、材料管理、实训成绩管理，以及使用传统方式点名签到已远远不能满足现代人为了节约时间、方便快捷的高效需求。开发一个校企合作实训平台，以多样化的管理方式为教学管理人员、企业教师、学生提供帮助。教师可在实训平台上统一对学生提交上传的材料进行下载管理，传统的点名签到工作量由漫长的教师点到时间、每位学生应答时间变成了每位同学的登录实训平台系统、进行签到工作的时间。而请假工作也不再需要繁琐的流程手续，也不再需要经过太多领导、老师签字批准，但是最基本的教师审核请假操作还是需要的。老师必须在实训平台系统中审核，写上审核批准请假通过原因或不允许请假的原因，再将审核状态由审核中改为不通过或已通过即可。诸多的管理功能极大地方便了企业教师、教学管理人员、学生在实训前、实训中、实训后参与实训，处理实训的相关事宜。

## 1.2 系统目标

在越来越多学校企业合作实习、培训的今天，为了方便企业教师管理学生，为了方便教学管理人员管理企业教师和学生，所以我们的软件校企合作实训平台应运而生。将企业地点定位在软件有限公司，学校定位是全各大高校，提供方便快捷、简单操作的实训平台，为各大高校与软件有限公司之间的合作提供服务。

设计成果：校企合作实训平台包括三个用户端，系统管理员端、企业教师端、学生端。

系统管理员端包括用户、主题、教师教学任务、实训前材料、消息管理等模块；用户管理包含添加、删除用户、给教师学生修改用户信息。主题管理包含主要基本功能。教师教学任务管理包括浏览、改、撤销任务、发布任务功能。实训前材料管理包括添加上传材料、删除材料、下载材料功能。消息管理包括发送、浏览、删除功能。

教师端包括项目管理、团队管理、阶段管理、参考资料管理、提交材料管理、消息管理、查看考勤表、成绩评定管理。项目管理包括基本操作和发布项目功能。团队管理包括组建学生团队、基本操作功能。阶段管理包括基本操作、阶段评审、进度监控功能。 参考资料管理包括（上传、下载、删）功能。 提交材料管理包括 （下载任务材料、打回）功能。消息管理包括（发送、浏览、删除）功能。成绩评定管理包括打分、查询、修改成绩功能。

学生端包括下载参考资料，提交上传材料，发送、浏览、删除消息，考勤管理（登录签到、请假、查看考勤表）功能，查看团队信息，查看项目信息，查看阶段信息，查看项目成绩等功能。

# 2 需求分析

## 2.1 业务需求

（1）教学管理人员、企业教师和学生可以在系统上进行登录、注册。系统管理员端包括用户管理、班级信息管理、主题管理、教师教学任务管理、实训前材料管理、消息管理等模块。用户管理包含添加、删除用户、给教师学生修改用户信息。主题管理包含主要基本操作。教师教学任务管理包括浏览、改、撤销、发布任务功能。实训前材料管理包括添加上传、删除、下载材料功能。消息管理包括发送、浏览、删除功能。

（2）教师端包括项目管理、团队管理、阶段管理、参考资料管理、提交材料管理、消息管理、查看考勤表、成绩管理。项目管理包括基本操作和发布项目功能。团队管理包括组建学生团队、基本操作功能。阶段管理包含基本操作功能。 参考资料管理包括（上传、下载、删）功能。 提交材料管理包括 （下载任务材料、打回）功能。消息管理包括（发送、浏览、删除）功能。成绩管理包括打分、查询、修改成绩功能。

（3）学生端包括下载参考资料，提交上传材料，发送、浏览、删除消息，考勤管理（登录签到、请假、查看考勤表）功能，查看团队信息，查看项目信息，查看阶段信息，查看项目成绩等功能。

（4）响应迅速，较为安全。

（5）系统稳定，并且应易于维护。

### 2.1.1主要业务流程

（1）查询用户信息活动图如图2-1所示：

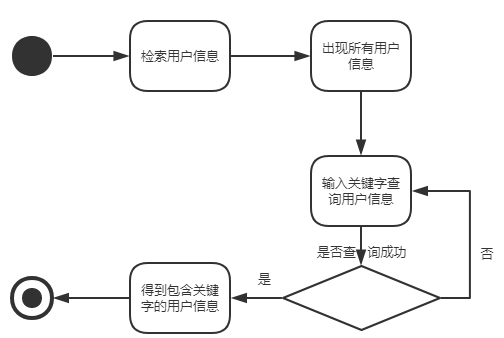


图2-1 查询用户信息活动图

（2）查询主题信息活动图如图2-2所示：

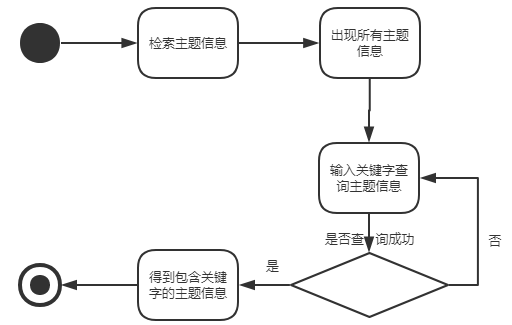


图2-2 查询主题信息活动图

（3）查询任务信息活动图如图2-3所示：

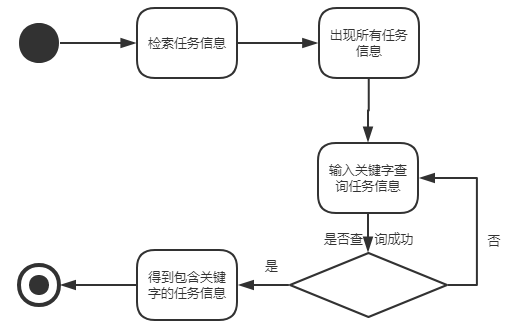


图2-3 查询任务信息活动图

（4）删除实训前材料活动图如图2-4所示：

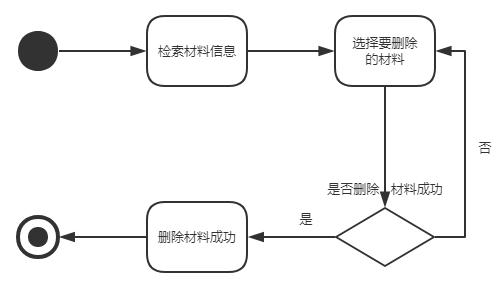


图2-4 删除实训前材料活动图

（5）查询团队信息活动图如图2-5所示：

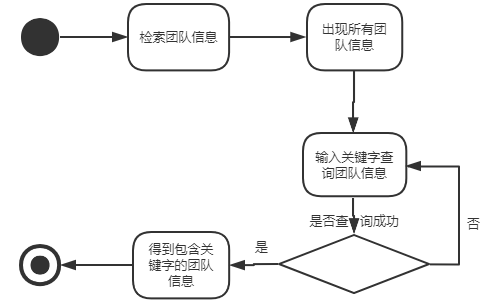


图2-5 查询团队信息活动图

（6）查询项目信息活动图如图2-6所示：

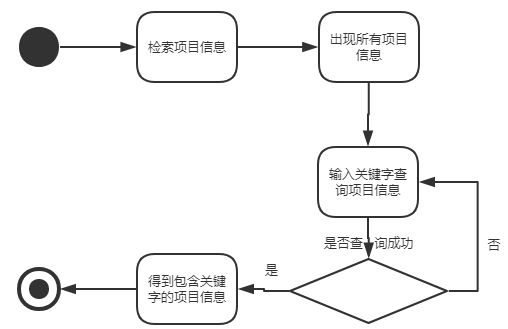


图2-6 查询项目信息活动图

（7）查询阶段信息活动图如图2-7所示：

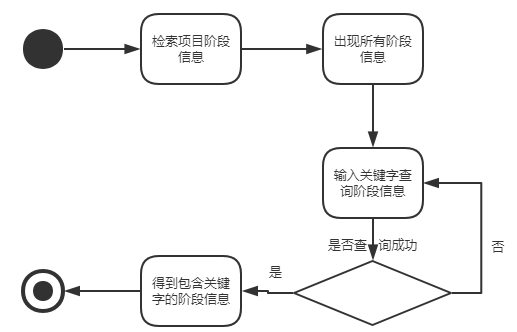


图2-7 查询阶段信息活动图

（8）删除参考资料活动图如图2-8所示：

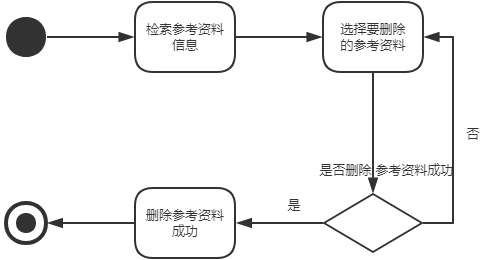


图2-8 删除实训前材料活动图

（9）删除提交材料活动图如图2-9所示：

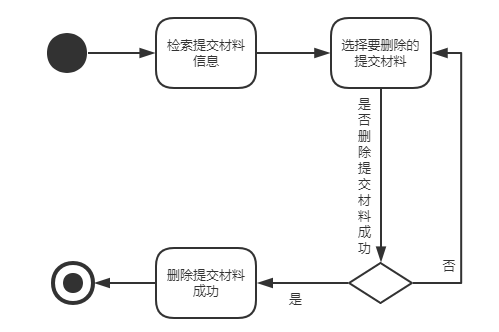


图2-9 删除提交材料活动图

（10）删除消息活动图如图2-10所示：

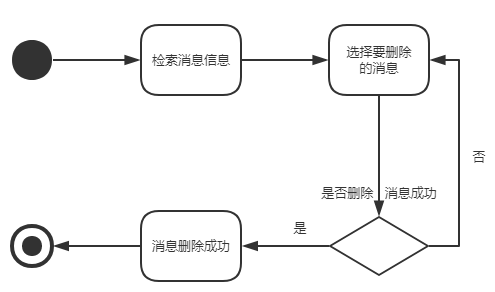


图2-10 删除消息活动图

（11）请假活动图如图2-11所示：

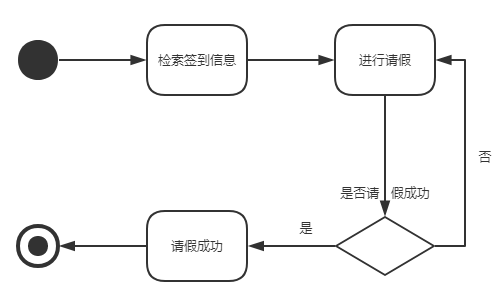


图2-11 请假活动图

（12）打分活动图如图2-12所示：

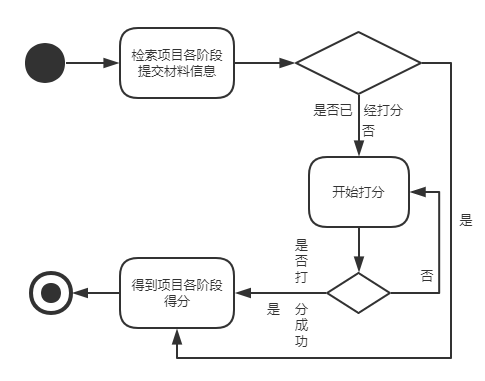


图2-12 打分活动图

## 功能需求

实训平台有实训前、实训中、实训后管理等模块。

实训前管理模块，该模块有用户管理、主题管理、任务管理、实训前材料管理，可以进行增删用户，添加修改用户信息，增删改主题，浏览主题内容等功能，上传、删除、下载实训前材料，可以进行修改、浏览任务信息，发布、撤销任务等功能。

实训中管理模块，该模块有项目管理、团队管理、阶段管理、参考资料管理、提交材料管理、消息管理、考勤管理，可以进行修改、浏览项目信息，发布、撤销项目，组建学生团队，修改、删除、浏览团队信息，发送、浏览、删除消息，签到、请假、查看考勤记录等功能。

实训后管理子系统，该模块主要包括成绩管理，可以进行成绩填写、成绩修改、删除成绩、查询成绩等功能。

### 2.2.1 角色分析

从校企合作实训平台的实际分析，平台有以下角色，得到系统角色表如表2-1所示：

表2-1 系统角色表

|  |  |
| --- | --- |
| 角色 | 职责或功能 |
| 系统管理员 | 对实训前的用户、主题、教师教学任务、实训前材料和实训中消息进行管理 |
| 教师 | 对实训前任务和材料以及实训中项目、团队、阶段、参考资料、提交材料、消息和实训后成绩进行管理 |
| 学生 | 对实训中的材料、消息、考勤、团队、阶段、项目信息和实训后的项目成绩进行管理 |

### 2.2.2 业务功能

从业务角度出发，给出了平台总体用例图，有登录注册、用户管理、主题管理、任务管理、实训前材料管理、项目管理、团队管理、阶段管理、参考资料管理、提交材料管理、消息管理、考勤管理、成绩管理等用例，如图2-13所示：

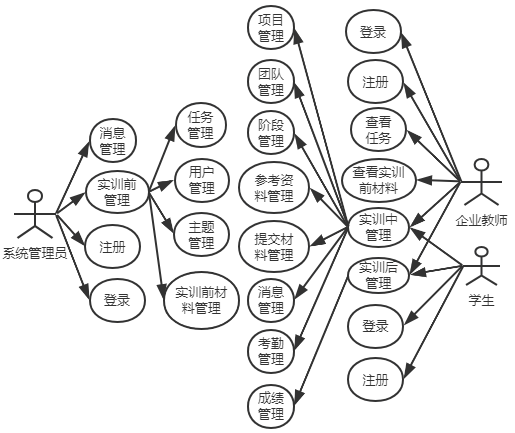


图2-13 系统总体用例图

（1）登录注册用例图如图2-14所示：

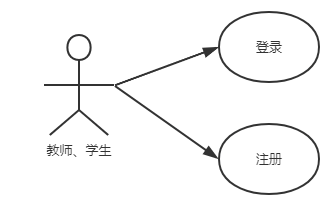


图2-14 登录注册用例图

登录用例描述如表2-2所示：

表2-2 登录用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 001 | 用例名称 | 登录 |
| 用例描述 | 登录是平台进行身份验证的功能模块，用户在提供有效的账号和密码后，发送系统后台进行身份验证，返回验证后的信息，根据返回信息判断身份验证是否通过。 | | |
| 执行者 | 管理员，教师，学生。 | | |
| 前置条件 | 进入登录界面，输入账号密码验证码 | | |
| 后置条件 | 登录成功，进入实训平台 | | |
| 涉众利益 | 1.使用者担心密码泄露  2.用户希望登录成功后能够记住密码 | | |
| 用例场景 | 系统登录界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1. 填写账号、密码及验证码  2. 系统验证账号、密码及验证码是否填写完整  3. 如果验证不完整，提示用户按要求填写必要的信息  4. 将有效的用户账号、密码及验证码提交到服务器验证  5. 系统根据服务器验证结果进行相应操作  6. 如果验证是验证码错误，直接提示重新输入验证码  7. 如果是账号或密码操作，直接提示账号或密码有误，不可以直接提示账号错误或密码错误  8. 如果验证通过，直接登录到系统主界面。  **扩展流程（Extend Flow）**  1a. 登录信息不充分：    1a1. 系统提示登录信息不充分 | | |

续表2-2

|  |  |
| --- | --- |
|  | 1a2. 用例结束  2a. 验证码错误：    2a1. 提示重新输入  2a2. 用例结束 |

注册用例描述如表2-3所示：

表2-3 注册用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 002 | 用例名称 | 注册 |
| 用例描述 | 注册是系统进行添加用户的功能模块，用户在提供有效的个人信息后，发送系统后台进行身份验证，返回验证后的信息，根据返回信息判断注册是否通过。 | | |
| 执行者 | 教师，学生。 | | |
| 前置条件 | 进入注册界面，输入个人信息 | | |
| 后置条件 | 注册成功，回到登录界面 | | |
| 涉众利益 | 1.使用者担心个人信息泄露 | | |
| 用例场景 | 系统注册界面 | | |
| 基本事件流 | **基本流程（Base Flow）**  1.填写个人信息  2.系统验证个人信息是否填写完整  3.如果不完整，提示用户按要求填写必须填写的信息  4.将有效的用户账号、密码及身份提交到服务器验证  5.系统根据服务器验证结果进行相应操作  6.如果验证不通过，直接提示重新输入个人信息  7.如果验证通过，直接回到登录界面。  **扩展流程（Extend Flow）**  1a. 注册信息不充分：    1a1.系统提示注册信息不充分，账号密码姓名身份必填项未填  1a2. 用例结束  2a. 验证不通过：  2a1. 提示账号重复重新输入  2a2. 用例结束 | | |

（2）用户管理用例图

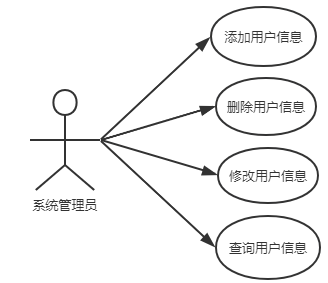


图2-15 用户管理用例图

添加用户信息用例描述如表2-4所示：

表2-4 添加用户信息用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 003 | 用例名称 | 添加用户信息 |
| 用例描述 | 描述了系统管理员在用户管理时添加不同用户的操作 | | |
| 执行者 | 系统管理员 | | |
| 前置条件 | 管理员进入用户管理页面 | | |
| 后置条件 | 管理员添加用户信息成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统用户管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 系统管理员登录系统成功，进入用户管理界面 2. 点击添加按钮 3. 输入用户信息 4. 输入完成，点击确认按钮，成功添加用户信息   **扩展流程（Extend Flow）**  1a. 添加用户信息不充分：    1a1.填写用户信息不充分，账号密码姓名身份必填项未填  1a2. 用例结束 | | |

（3）主题管理用例图

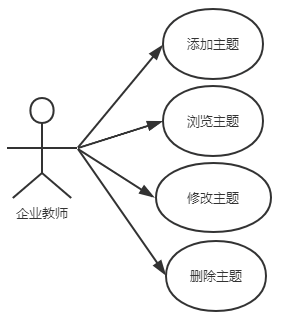


图2-16 主题管理用例图

浏览主题用例描述如表2-5所示：

表2-5 浏览主题用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 004 | 用例名称 | 浏览主题 |
| 用例描述 | 描述了系统管理员在主题管理时查询浏览主题的操作 | | |
| 执行者 | 系统管理员 | | |
| 前置条件 | 管理员进入主题管理页面 | | |
| 后置条件 | 管理员浏览主题成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统主题管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 系统管理员登录系统成功，进入主题管理界面 2. 浏览全部主题内容 3. 在文本框中输入关键字查询主题信息 4. 点击查询按钮   成功查询浏览主题信息  **扩展流程（Extend Flow）**  1a. 查询的主题信息不存在：    1a1. 查询到的主题信息为空  1a2. 用例结束 | | |

（4）任务管理用例图

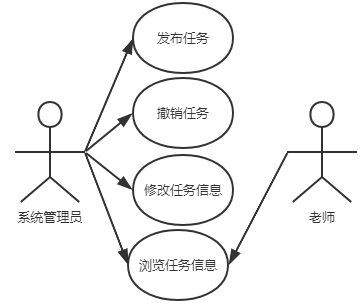


图2-17 任务管理用例图

修改任务信息用例描述如表2-6所示：

表2-6 修改任务信息用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 005 | 用例名称 | 修改任务信息 |
| 用例描述 | 描述了系统管理员在教师教学任务管理时修改任务信息的操作 | | |
| 执行者 | 系统管理员 | | |
| 前置条件 | 管理员进入教师教学任务管理页面 | | |
| 后置条件 | 管理员修改任务信息成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统教师教学任务管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 系统管理员登录系统成功，进入教师教学任务管理界面 2. 选择要修改的任务，点击修改按钮 3. 编辑任务信息 4. 点击确认按钮，成功修改任务信息   **扩展流程（Extend Flow）**  1a. 编辑任务信息不充分：  1a1.填写任务信息不充分，编号名称必填项未填  1a2.用例结束 | | |

（5）实训前材料管理用例图

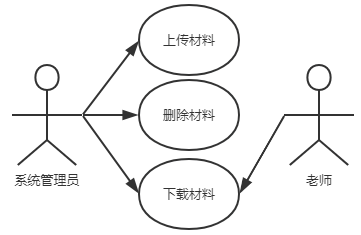


图2-18 实训前材料管理用例图

删除材料用例描述如表2-7所示：

表2-7 删除材料用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 006 | 用例名称 | 删除材料 |
| 用例描述 | 描述了系统管理员在实训前材料管理时删除材料的操作 | | |
| 执行者 | 系统管理员 | | |
| 前置条件 | 管理员进入实训前材料管理页面 | | |
| 后置条件 | 管理员删除材料成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统开班前材料管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 系统管理员登录系统成功，进入实训前材料管理界面 2. 点击删除按钮 3. 选择要删除的材料 4. 点击确认按钮，成功删除材料 | | |

（6）项目管理用例图

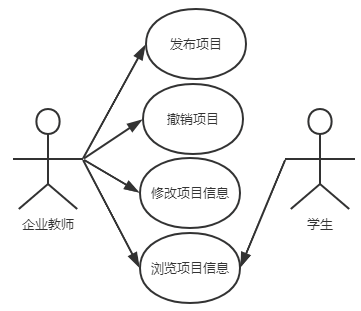


图2-19 项目管理用例图

发布项目信息用例描述如表2-8所示：

表2-8 发布项目信息用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 007 | 用例名称 | 发布项目信息 |
| 用例描述 | 描述了企业教师在项目管理时发布项目信息的操作 | | |
| 执行者 | 企业教师 | | |
| 前置条件 | 教师进入项目管理页面 | | |
| 后置条件 | 教师发布项目信息成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统项目管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 教师登录系统成功，进入项目管理界面 2. 点击发布按钮 3. 输入项目信息 4. 输入完成，点击确认按钮，成功发布项目信息   **扩展流程（Extend Flow）**  1a. 发布项目信息不充分：    1a1.填写任务信息不充分，编号名称状态所属团队必填项未填  1a2.用例结束 | | |

（7）团队管理用例图

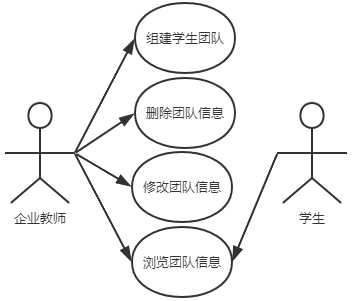


图2-20 团队管理用例图

组建学生团队用例描述如表2-9所示：

表2-9 组建学生团队用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 008 | 用例名称 | 组建学生团队 |
| 用例描述 | 描述了企业教师在团队管理时组建学生团队的操作 | | |
| 执行者 | 企业教师 | | |
| 前置条件 | 教师进入团队管理页面 | | |
| 后置条件 | 教师组建添加团队成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统团队管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 教师登录系统成功，进入团队管理界面 2. 点击添加按钮 3. 输入团队信息 4. 输入完成，点击确认按钮，成功组建学生团队   **扩展流程（Extend Flow）**  1a. 添加团队信息不充分：    1a1.填写团队信息不充分，编号名称必填项未填  1a2.用例结束 | | |

（8）阶段管理用例图

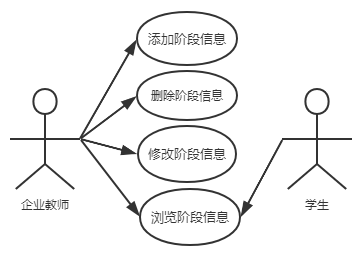


图2-21 阶段管理用例图

添加阶段信息用例描述如表2-10所示：

表2-10 添加阶段信息用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 009 | 用例名称 | 添加阶段信息 |
| 用例描述 | 描述了企业教师在阶段管理时添加阶段信息的操作 | | |
| 执行者 | 企业教师 | | |
| 前置条件 | 教师进入阶段管理页面 | | |
| 后置条件 | 教师添加阶段信息成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统阶段管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 教师登录系统成功，进入阶段管理界面 2. 点击添加按钮 3. 输入阶段信息 4. 输入完成，点击确认按钮，成功添加阶段信息   **扩展流程（Extend Flow）** | | |
|  | 1a. 添加阶段信息不充分：    1a1.填写阶段信息不充分，编号名称所属项目所属者必填项未填  1a2.用例结束 | | |

（9）参考资料管理用例图

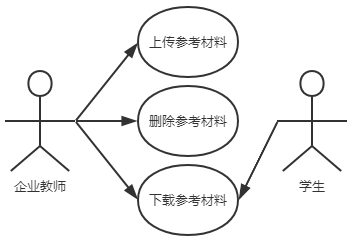


图2-22 参考资料管理用例图

上传参考资料用例描述如表2-11所示：

表2-11 上传参考资料用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 010 | 用例名称 | 上传参考资料 |
| 用例描述 | 描述了教师在参考资料管理时上传参考资料的操作 | | |
| 执行者 | 教师 | | |
| 前置条件 | 教师进入参考资料管理页面 | | |
| 后置条件 | 教师上传参考资料成功 | | |
| 涉众利益 | 说明涉众及涉众关心和担心的事情。  1.教师担心上传的参考资料会被外泄 | | |
| 用例场景 | 系统参考资料管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 教师登录系统成功，进入参考资料管理界面 2. 点击上传按钮 3. 选择要上传的文件 4. 点击确认按钮，成功上传材料   **扩展流程（Extend Flow）**  无 | | |

（10）提交材料管理用例图

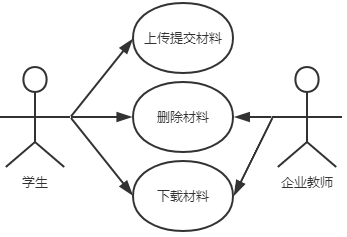


图2-23 提交材料管理用例图

下载材料用例描述如表2-12所示：

表2-12 下载材料用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 011 | 用例名称 | 下载材料 |
| 用例描述 | 描述了用户在提交材料管理时下载材料的操作 | | |
| 执行者 | 教师、学生 | | |
| 前置条件 | 用户进入提交材料管理页面 | | |
| 后置条件 | 用户下载材料成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统提交材料管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 用户登录系统成功，进入提交材料管理界面 2. 点击下载按钮 3. 选择要下载的材料 4. 选择要保存的位置 5. 点击确认按钮，成功下载材料   **扩展流程（Extend Flow）**  无 | | |

（11）消息管理用例图

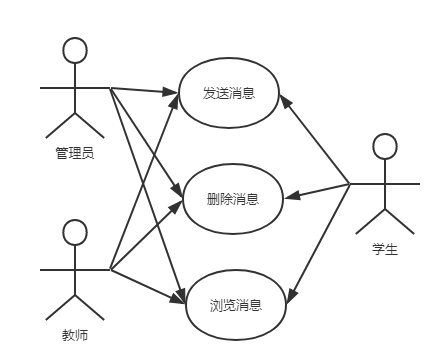


图2-24 消息管理用例图

发送消息用例描述如表2-13所示：

表2-13 发送消息用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 012 | 用例名称 | 发送消息 |
| 用例描述 | 描述了用户在消息管理时发送消息的操作 | | |
| 执行者 | 管理员、企业教师、学生 | | |
| 前置条件 | 用户进入消息管理页面 | | |
| 后置条件 | 用户发送消息成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统消息管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）** | | |
|  | 1. 用户登录系统成功，进入消息管理界面 2. 点击发送按钮 3. 输入消息内容 4. 输入完成，点击确认按钮，成功发送消息   **扩展流程（Extend Flow）**  1a. 发送消息不充分：    1a1.消息内容不能为空  1a2.用例结束 | | |

（12）考勤管理用例图

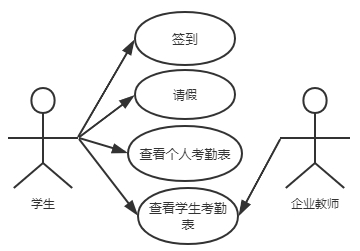


图2-25 考勤管理用例图

签到用例描述如表2-14所示：

表2-14 签到用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 013 | 用例名称 | 签到 |
| 用例描述 | 描述了学生在考勤管理时签到的操作 | | |
| 执行者 | 学生 | | |
| 前置条件 | 学生进入考勤管理页面 | | |
| 后置条件 | 学生签到成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统考勤管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 学生登录系统成功，进入考勤管理界面 2. 点击签到按钮 3. 点击确认按钮 4. 界面显示签到信息，成功签到   **扩展流程（Extend Flow）**  无 | | |

（13）成绩管理用例图

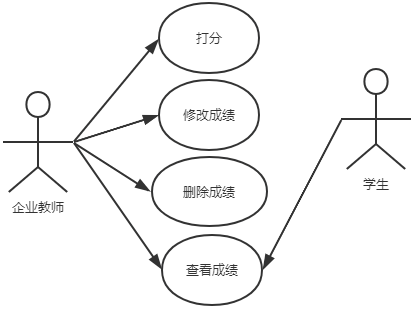


图2-26 成绩管理用例图

打分用例描述如表2-15所示：

表2-15 打分用例描述

|  |  |  |  |
| --- | --- | --- | --- |
| 用例编号 | 014 | 用例名称 | 打分 |
| 用例描述 | 描述了教师在成绩管理时打分的操作 | | |
| 执行者 | 教师 | | |
| 前置条件 | 教师进入成绩管理页面 | | |
| 后置条件 | 教师打分成功 | | |
| 涉众利益 | 无 | | |
| 用例场景 | 系统成绩管理界面 | | |
| 基本事件流 | **基本流程（Base Flow）**   1. 教师登录系统成功，进入成绩管理界面 2. 查看学生项目是否已完成所有阶段 3. 对已完成所有阶段的项目打分，点击打分按钮 4. 输入分数 5. 点击确认按钮，成功打分   **扩展流程（Extend Flow）**  1a. 打分错误：    1a1. 系统提示项目未完成所有阶段，无法打分  1a2. 用例结束 | | |

## 2.3 非功能需求

### 2.3.1环境需求

服务器端环境

表 2-48服务器端环境

|  |  |
| --- | --- |
| 系统所使用数据库 | Mysql 8.0 |
| 操作系统 | Windows 10 专业版 |
| Web服务器 | Tomcat 8.0.411 |

客户端环境

表 2-49客户端环境

|  |  |
| --- | --- |
| 操作系统 | Windows 7及其以上版本的操作系统 |
| 浏览器 | Firefox、Chrome等 |
| 分辨率 | 最佳效果为1920×1080像素 |

### 2.3.2 性能需求

1. 可维护性

维护方法：平台能做到使在数据损坏或丢失等情况下，将之前备份好的数据重新找回，实现数据恢复。

1. 易操作性

界面设计：平台应提供实用，方便、直接的图形用户管理界面，简单易学，平均响应时间应控制在1-3s内，操作方便，所有菜单驱动的处理和各种快捷键，一键功能以确保多数达到。

1. 可扩展性

在实训平台设计时，应当考虑到日后可能新增的扩展功能模块，提供扩展接口，保证具有一定的扩展能力。

软硬件升级：系统应采取的硬件和软件平台，软件和硬件的负载平衡机制的可扩展性充分考虑。

### 2.3.3 安全需求

1. 容错性

在出现错误的时候系统能够及时中止错误行为，弹出友好的信息提示，并对已经出现的错误进行操作日志记录，确保之后系统管理员能追踪出错原因。

1. 系统外部环境

本系统针对部署的外部环境，进行了网络安全防护措施，开启系统备份、防火墙和用户权限认证，使操作系统和应用系统处于安全良好的网络环境下，减少来自外部网络的威胁。

1. 系统内部安全

该实训平台还确保不同用户使用到的界面、功能是不一样的，保障实训平台数据的内部安全。

1. 系统运行安全

从逻辑上讲，该系统应具有抵抗非法入侵的能力；在物理方面，该系统应确保没有潜在的单点故障，并提供资源的数据备份功能。还能通过mysql的备份日志在数据损坏或丢失的情况下进行数据的重建。

3 总体设计

## 3.1系统设计的原则

1. 实用性原则。实用性原则可以说决定了使用该实训平台经营效益的高与低。因此在进行该系统的设计时，我将系统的实用性要求放在了首位。本系统力求能够紧密切合与学校合作的企业和学生、教师间的实际业务需求，为了不再出现学校与企业之间合作实训不方便管理的问题。
2. 先进性原则。除了上述实用性以外，还不得不考虑实训平台的先进性。因此，为了紧跟时代的脚步，平台能有个更好的发展，在本校企合作实训平台的设计过程中我采用目前较为先进的java框架，使实训平台各方面都优于过去。
3. 安全性原则。实训平台的安全机制为了不出现数据流失、错误显示等状况，为各个用户端提供准确的用户使用不同功能的权限划分，对于数据库管理，建立起完备的数据备份恢复机制。
4. 经济可行性原则。本系统在满足系统功能及性能要求下，通过采用开源的SSM技术架构和Tomcat应用服务器及Mysql数据库服务器进行系统设计，以降低系统的搭建和维护成本，达到良好的经济可行性。
5. 可扩展性原则。本系统采取规范的数据接口命名和实现方式，遵循开放性原则的设计，支持二次开发，可以在日后对其功能进行扩展。因为学生、企业教师等对实训平台的需求会随技术而变化。

## 系统体系结构设计

软件校企合作实训平台采用的是ssm框架，以下是关于ssm框架的简单介绍：

（1）背景介绍

SSM框架，是Spring + Spring MVC + MyBatis的缩写，Spring依赖注入DI来管理各层的组件，使用面向切面编程AOP管理事物、日志、权限等。SpringMVC代表了Model(模型)View(视图)Controller(控制)接收外部请求,进行分发和处理。Mybatis是基于jdbc的框架,主要用来操作数据库,并且将业务实体和数据表联系起来。

（2）知识剖析SSM整合思路

spring在进行管理时，是很有条理的，每个层都由spring管理，然后不同的层可以调用其它层，Handler调用service，service调用mapper等。整合dao层。mybatis和spring整合，通过spring管理mapper接口。使用mapper的扫描器自动扫描mapper接口在spring中进行注册。整合service层。通过spring管理 service接口。使用配置方式将service接口配置在spring配置文件中。实现事务控制。整合springmvc。由于springmvc是spring的模块，不需要整合。

1. SpringMVC流程

服务器发送Http request请求，请求被前端控制器（DispatcherServlet）捕获。

前端控制器根据xml文件中的配置（或者注解）对请求的URL进行解析，得到请求资源标识符（URI）。然后根据该URI，调用处理器映射器（HandlerMapping）获得处理该请求的Handler以及Handler对应的拦截器，最后以 HandlerExecutionChain 对象的形式返回。前端控制器根据获得的Handler，选择一个合适的处理器适配器（HandlerAdapter）去执行该Handler。处理器适配器提取request中的模型数据，填充Handler入参，执行处理器（Handler）（也称之为Controller）。Handler(Controller)执行完成后，向处理器适配器返回一个ModelAndView对象，处理器适配器再向前端控制器返回该ModelAndView对象（ModelAndView只是一个逻辑视图）。根据返回的ModelAndView，前端控制器请求一个适合的视图解析器（ViewResolver）（必须是已经注册到Spring容器中的ViewResolver）去进行视图解析，然后视图解析器向前端控制器返回一个真正的视图View（jsp）。前端控制器通过Model解析出ModelAndView中的参数进行解析，最终展现出完整的View并通过Http response返回给客户端。

本平台通过对业务流程整理和分析，得了出实训平台的需求。在需求分析的基础上，又进行系统设计，明确了在B/S模式下采用三层架构来实现目标系统。又进行了系统的数据库设计，其中有实体关系（E-R图）和逻辑设计。最终，对系统进行了开发和运行测试，通过对SSM框架技术的运用，开发出了一个数据交互良好、实用性好、且能满足当前需求的校企合作实训平台。

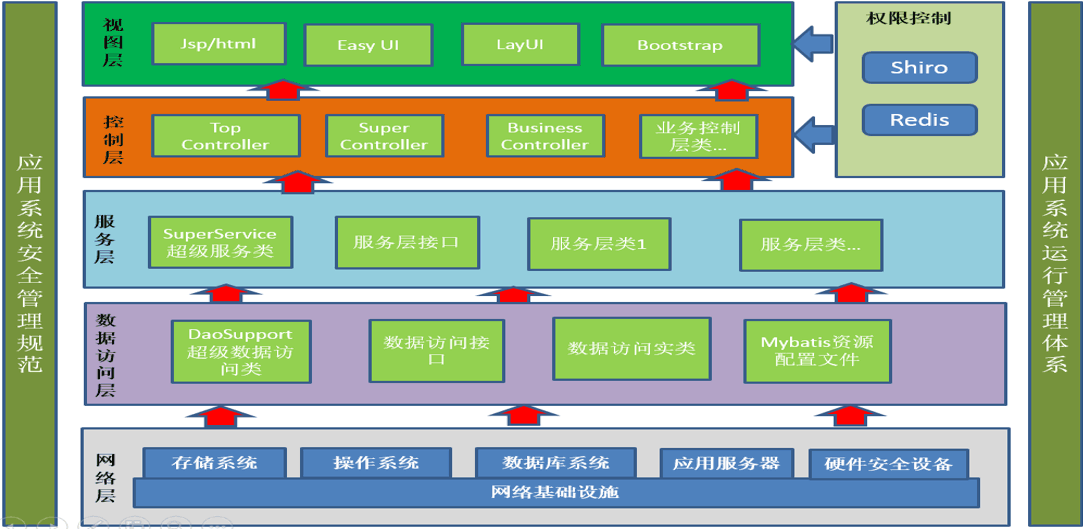


图3-1 软件校企合作实训平台系统结构图

## 3.3 系统功能结构设计

本系统主要包括实训前管理、实训中管理、实训后管理几大子系统，如下图3-2所示：

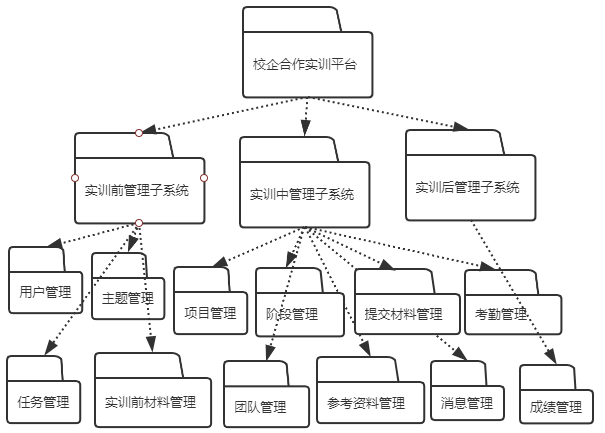


图3-2 系统总包图

1. 实训前管理子系统：

该子系统主要包括用户管理、主题管理、任务管理、实训前材料管理，可以进行增删用户，修改用户信息，增删改主题，浏览主题内容等功能，其详细的用例说明见《系统需求说明书》。

1. 实训中管理子系统：

该子系统主要包括项目管理、团队管理、阶段管理、参考资料管理、提交材料管理、消息管理、考勤管理，可以进行查询项目信息，发布项目等功能，其详细的用例说明见《系统需求说明书》。

1. 实训后管理子系统：

该子系统主要包括成绩管理，可以进行打分、成绩修改，添加、修改，其详细的用例说明见《系统需求说明书》。

# 4 数据据库设计

## 4.1 概念结构设计

### 4.1.1 设计思路

该系统是校企合作实训平台，其中一个主题可以有多个任务，一个任务可以有多个实训前材料，一个任务可以有多个项目，一个团队只有一个项目，一个项目可以有多个阶段，一个项目可以有多个参考资料，一个阶段有多个提交材料，一个提交材料对应一个成绩。实体有学生实体、教师实体、主题实体、任务实体、材料实体、团队实体、项目实体、阶段实体、签到实体、请假实体、成绩实体。

再通过功能需求来确定实体的属性：

1. 学生实体，属性有学号、密码、姓名、班级、团队名称、头像、性别、电话、备注。
2. 教师实体，属性包括工号、密码、姓名、班级、头像、性别、电话、备注。
3. 主题实体，介绍本次实训的主题内容。属性包括序号、主题、添加时间。
4. 任务实体，介绍本次实训任务的内容。属性包括任务编号、任务名称、教师、主题、添加时间、发布人。
5. 实训前材料实体，属性包括序号、文件名称、任务编号、教师，主题、添加人，添加时间。
6. 团队实体，属性包括团队编号、团队名称、团队图片、队长、带班老师、口号。
7. 项目实体，属性包括项目名称、任务编号、主题、团队名称、图片、发布老师、完成状态。
8. 阶段实体，属性包括序号、项目名称、团队名称、阶段名称、满分、学生、带班老师、完成状态。
9. 参考资料实体，介绍与项目有关的材料信息，属性包括序号、文件名称、项目名称、阶段名称、团队名称、添加人、添加时间。
10. 提交材料实体，介绍学生上传的材料信息，属性包括序号、文件名称、项目名称、阶段名称、审批教师、提交学生、满分、打分状态。
11. 消息实体，属性包括发送人、消息内容、接收人、时间。
12. 签到实体，属性包括序号、签到备注、签到人、签到时间。
13. 成绩实体，属性包括序号、项目名称、阶段名称、团队名称、学生、满分、得分、打分老师。
14. 管理员实体，属性包括序号、用户名、密码、添加时间。
15. 班级实体，属性包括序号、班级、添加时间。
16. 请假实体，属性包括序号、开始时间、结束时间、请假理由、审批状态、请假人、请假时间。

### 4.1.2 E-R图

（1）各实体关系E-R图

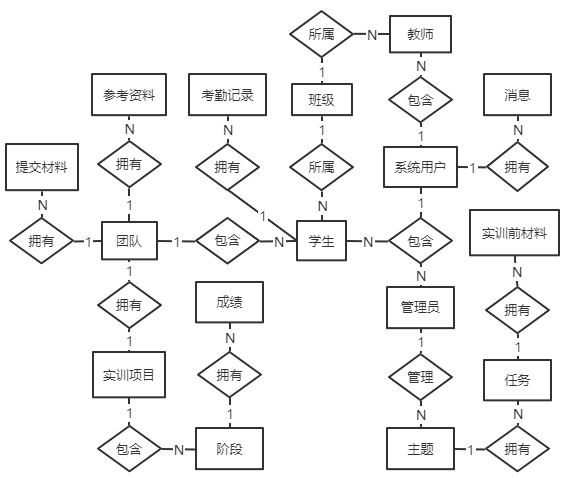


图4-1：各实体关系E-R图

（2）学生信息实体图

实体：学生

属性：学号，密码，姓名，班级，团队名称，头像，性别，电话，备注。

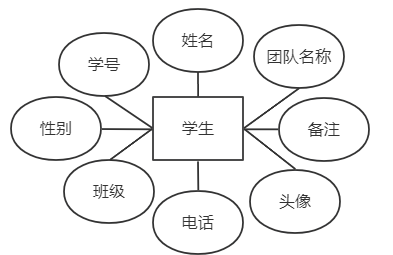


图4-2- 学生实体图

（3）任务实体图

实体：任务

属性：任务编号，任务名称，教师，主题，添加时间，发布人。

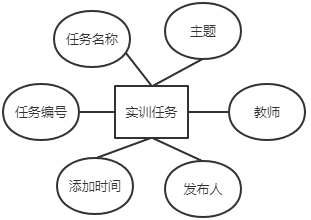


图4-3 任务实体图

（4）实训前材料实体图

实体：实训前材料

属性：序号，文件名称，任务编号，教师，主题，添加人，添加时间。

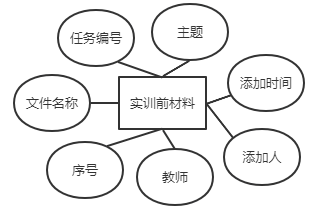


图4-4 实训前材料实体图

（5）实训项目实体图

实体：项目

属性：项目名称，任务编号，主题，团队名称，图片，发布老师，完成状态。

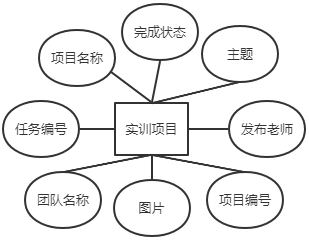


图4-5 实训项目实体图

（6）团队实体图

实体：团队

属性：团队编号，团队名称，团队图片，队长，带班老师，口号。

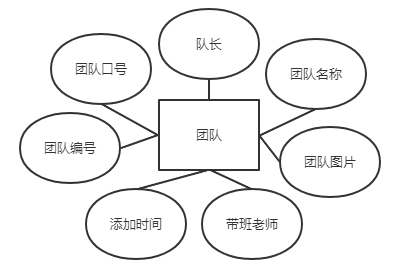


图4-6 团队实体图

（7） 实训阶段实体图

实体：阶段

属性：序号，项目名称，团队名称，阶段名称，满分，学生，带班老师，完成状态。

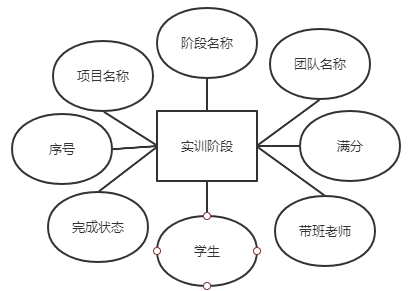


图4-7 实训阶段实体图

（8）参考资料实体图

实体：参考资料

属性：序号，文件名称，项目名称，阶段名称，团队名称，添加人，添加时间。

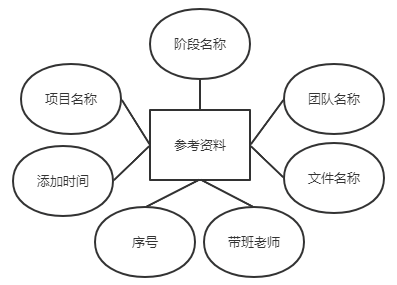


图4-8 参考资料实体图

（9）提交材料实体图

实体：提交材料

属性：序号，文件名称，项目名称，阶段名称，审批教师，提交学生，满分，打分状态。

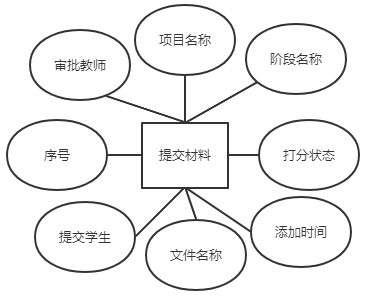


图4-9 提交材料实体图

（10）消息实体图

实体：消息

属性：发送人，消息内容，接收人，时间。

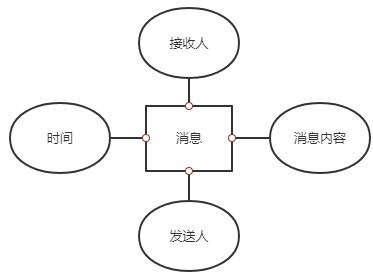


图4-10 消息实体图

（11）学生签到记录实体图

实体：签到

属性：序号，签到备注，签到人，签到时间

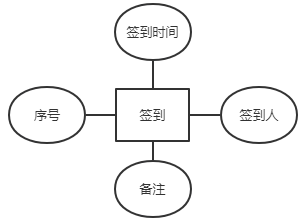


图4-11学生签到记录实体图

（12）学生成绩实体图

实体：成绩

属性：序号，项目名称，阶段名称，团队名称，学生，满分，得分，打分老师。

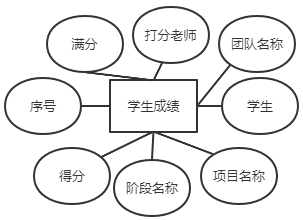


图4-12 学生成绩实体

（13）实训主题图

实体：主题

属性：序号，主题，添加时间

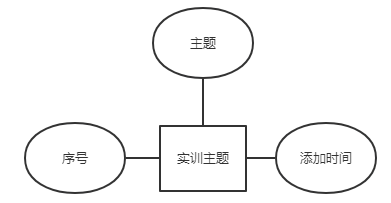


图4-13 实训主题实体

（14）教师信息实体图

实体：教师

属性：工号，密码，姓名，班级，头像，性别，电话，备注。

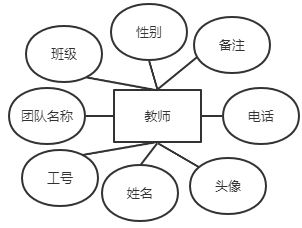


图4-14 教师信息实体

（15）管理员信息实体图

实体：管理员

属性：序号，用户名，密码，添加时间。

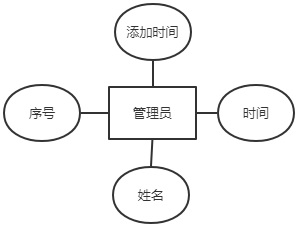


图4-15 管理员信息实体

（16）班级信息实体图

实体：班级

属性：序号，班级，添加时间。

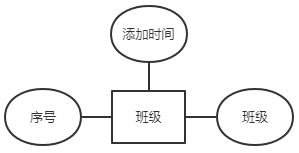


图4-16 管理员信息实体

（17）请假记录实体图

实体：请假

属性：序号，开始时间，结束时间，请假理由，审批状态，请假人，请假时间。

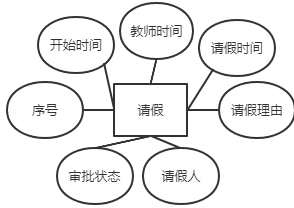


图4-17 管理员信息实体

## 4.2 逻辑结构设计

数据库的逻辑结构设计就是把所有的实体-关系图转化为数据库系统中相对应的逻辑结构。

### 4.2.1 设计思路

先根据功能需求确认到底有几个数据库实体，再根据各实体E-R图和各实体关系E-R图来确定各数据库实体的关系和它们有哪些属性，最后定义主键和外键，主键一般是ID编号，外键的定义要根据数据实体之间关系一对一，一对多或多对多的具体情况来定义。

4.2.2 逻辑模型

E-R图转换为关系模式

1. 学生表（序号，学号，密码，姓名，班级，团队名称，头像，性别，电话，备注）
2. 教师表（序号，工号，密码，姓名，班级，头像，性别，电话，备注）
3. 管理员表（序号，用户名，密码，添加时间）
4. 班级表（序号，班级，添加时间）
5. 主题表（序号，主题，添加时间）
6. 任务表（序号，任务编号，任务名称，教师，主题，添加时间，发布人）
7. 实训前材料表（序号，文件名称，任务编号，教师，主题，添加人，添加时间）
8. 团队表（序号，团队编号，团队名称，团队图片，队长，带班老师，口号）
9. 项目表（序号，项目名称，任务编号，主题，团队名称，图片，发布老师，完成状态）
10. 阶段表（序号，项目名称，团队名称，阶段名称，满分，学生，带班老师，完成状态）
11. 参考资料表（序号，文件名称，项目名称，阶段名称，团队名称，添加人，添加时间）
12. 提交材料表（序号，文件名称，项目名称，阶段名称，审批教师，提交学生，满分，打分状态）
13. 消息表（序号，发送人，消息内容，接收人，时间）
14. 签到表（序号，签到备注，签到人，签到时间）
15. 请假表（序号，开始时间，结束时间，请假理由，审批状态，请假人，请假时间）
16. 成绩表（序号，项目名称，阶段名称，团队名称，学生，满分，得分，打分老师）

## 4.3 物理结构设计

### 4.3.1 存取方式

通过分析软件校企合作实训平台的业务逻辑，设计出了数据库的逻辑结构，根据数据库的逻辑结构确定使用MySQL数据库存储数据，使用HashMap索引存取方法，因为MySQL数据库简单易用，它由多种程序和库组成，能够快捷、有效、安全的处理大量数据。

### 4.3.2 存储结构

学生表

表4-1 学生表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | Xingming | varchar | 20 | 否 | 姓名 |
| 3 | Xingbie | varchar | 2 | 否 | 性别 |
| 4 | Dianhua | varchar | 10 | 否 | 电话号码 |

续表4-1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 5 | Banji | Varchar | 11 | 否 | 实训班级编号，外键 |
| 6 | Tuandui | Varchar | 11 | 否 | 团队编号，外键 |
| 7 | mima | varchar | 50 | 否 | 密码 |
| 8 | Touxiang | Varchar | 255 | 否 | 头像 |
| 9 | Beizhu | Text | 0 | 否 | 备注 |
| 10 | Xuehao | Varchar | 50 | 否 | 学号 |

教师表

表4-2 教师表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | Xingming | varchar | 20 | 否 | 姓名 |
| 3 | Xingbie | varchar | 2 | 否 | 性别 |
| 4 | Dianhua | varchar | 10 | 否 | 电话号码 |
| 5 | Banji | Varchar | 11 | 否 | 实训班级编号，外键 |
| 6 | mima | varchar | 50 | 否 | 密码 |
| 7 | Touxiang | Varchar | 255 | 否 | 头像 |
| 8 | Beizhu | Text | 0 | 否 | 备注 |
| 9 | Gonghao | Varchar | 50 | 否 | 工号 |

管理员表

表4-3 管理员表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | Username | Varchar | 50 | 否 | 用户名 |
| 3 | Pwd | Varchar | 50 | 否 | 密码 |
| 4 | Addtime | Timestap | 0 | 否 | 添加时间 |

班级表

表4-4 班级表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 实训班级编号 |
| 2 | name | varchar | 20 | 否 | 实训班级名称 |
| 3 | Addtime | Timestap | 0 | 否 | 添加时间 |

主题表

表4-5 主题表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | Zhuti | varchar | 20 | 否 | 主题 |
| 3 | Addtime | Timestap | 0 | 否 | 添加时间 |

实训前任务表：

表4-6 实训前任务表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | Renwubianhao | int | 20 | 否 | 实训前任务编号 |
| 2 | name | varchar | 20 | 否 | 实训前任务名称 |
| 3 | Zhuti | int | 11 | 否 | 实训主题，外键 |
| 4 | id | int | 11 | 是 | 序号 |
| 5 | Jiaoshi | Varchar | 20 | 否 | 教师 |
| 6 | Tianjiaren | Varchar | 20 | 否 | 发布人 |
| 7 | Addtime | Timestap | 0 | 否 | 添加时间 |

实训前材料表

表4-7 实训前材料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | name | varchar | 30 | 否 | 文件名称 |

续表4-7

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3 | Renwubianhao | int | 20 | 否 | 实训前任务编号 |
| 4 | Zhuti | int | 11 | 否 | 实训主题，外键 |
| 5 | Jiaoshi | Varchar | 10 | 否 | 教师 |
| 6 | Tianjiaren | Varchar | 10 | 否 | 添加人 |
| 7 | Addtime | Timestap | 0 | 否 | 添加时间 |

团队表

表4-8 团队表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | tuanduibianhao | Varchar | 20 | 否 | 序号 |
| 2 | Tuanduimingcheng | Varchar | 10 | 否 | 团队名称 |
| 3 | Slogan | Varchar | 50 | 否 | 团队口号 |
| 4 | Id | Varchar | 11 | 是 | 序号 |
| 5 | Tuanduitupian | Varchar | 255 | 否 | 团队图片 |
| 6 | Captain | Varchar | 10 | 否 | 队长 |
| 7 | Tianjiaren | Varchar | 10 | 否 | 带班老师 |

实训项目表

表4-9 实训项目表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | Xiangmumingcheng | varchar | 20 | 否 | 项目名称 |
| 3 | Renwubianhao | int | 20 | 否 | 实训前任务编号 |
| 4 | Zhuti | int | 11 | 否 | 实训主题，外键 |
| 5 | Tuanduimingcheng | varchar | 10 | 否 | 团队名称 |
| 6 | state | varchar | 10 | 否 | 完成状态 |
| 7 | Tupian | varchar | 255 | 否 | 图片 |
| 8 | Laoshi | Varchar | 10 | 否 | 发布老师 |

实训阶段表

表4-10 实训阶段表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 实训阶段编号 |
| 2 | Jieduanmingcheng | varchar | 10 | 否 | 阶段名称 |
| 3 | Xiangmumingcheng | varchar | 20 | 否 | 项目名称 |
| 4 | Tuanduimingcheng | varchar | 10 | 否 | 团队名称 |
| 5 | state | varchar | 10 | 否 | 阶段完成状态 |
| 6 | Manfen | Int | 11 | 否 | 满分 |
| 7 | Xuesheng | Varchar | 10 | 否 | 学生 |
| 8 | Laoshi | Varchar | 10 | 否 | 带班老师 |

参考资料表

表4-11 参考资料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | name | varchar | 30 | 否 | 文件名称 |
| 3 | Tianjiaren | Varchar | 10 | 否 | 添加老师 |
| 4 | Jieduanmingcheng | varchar | 10 | 否 | 阶段名称 |
| 5 | Xiangmumingcheng | varchar | 20 | 否 | 项目名称 |
| 6 | Tuanduimingcheng | varchar | 10 | 否 | 团队名称 |
| 7 | Addtime | Timestap | 0 | 否 | 添加时间 |

提交材料表

表4-12 提交材料表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 序号 |
| 2 | name | varchar | 30 | 否 | 文件名称 |
| 3 | Chuangjianren | Varchar | 10 | 否 | 审批老师 |
| 4 | Jieduanmingcheng | varchar | 10 | 否 | 阶段名称 |
| 5 | Xiangmumingcheng | varchar | 20 | 否 | 项目名称 |

续表4-12

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 6 | Tijiaoren | Varchar | 10 | 否 | 提交学生 |
| 7 | Zhuangtai | varchar | 10 | 否 | 打分状态 |
| 8 | Manfen | Int | 11 | 否 | 满分 |

消息表

表4-13 消息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 消息编号 |
| 2 | Fasongren | varchar | 10 | 否 | 发送人 |
| 3 | content | text | 0 | 否 | 消息内容 |
| 4 | date | Timestap | 0 | 否 | 发送时间 |
| 5 | Jieshouren | varchar | 10 | 否 | 接收人 |

学生签到记录表

表4-14 学生签到记录表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 签到记录编号 |
| 2 | Qiandaobeizhu | Text | 0 | 否 | 签到备注 |
| 3 | Qiandaoren | Varchar | 10 | 否 | 签到人 |
| 4 | date | datetime | 0 | 否 | 签到时间 |

请假表

表4-15 请假表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 请假编号 |
| 2 | Kaishishijian | Varchar | 25 | 否 | 开始时间 |
| 3 | Jieshushijian | Varchar | 25 | 否 | 结束时间 |
| 4 | Qingjialiyou | Text | 0 | 否 | 请假理由 |
| 5 | Zhuangtai | Varchar | 10 | 否 | 状态 |

续表4-15

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 6 | Qingjiaren | Varchar | 10 | 否 | 请假人 |
| 7 | Addtime | Timestap | 0 | 否 | 请假时间 |

学生成绩表

表4-16 学生成绩表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 字段长度 | 是否主键 | 说明 |
| 1 | id | int | 11 | 是 | 成绩编号 |
| 2 | Manfen | int | 11 | 否 | 满分 |
| 3 | Defen | int | 11 | 否 | 得分 |
| 4 | Tuanduimingcheng | varchar | 10 | 否 | 团队名称 |
| 5 | Tianjiaren | Varchar | 10 | 否 | 打分老师 |
| 6 | Jieduanmingcheng | varchar | 10 | 否 | 阶段名称 |
| 7 | Xiangmumingcheng | varchar | 20 | 否 | 项目名称 |
| 8 | Tijiaoren | varchar | 10 | 否 | 学生 |

5 界面设计

## 界面关系图或工作流图

系统界面关系图

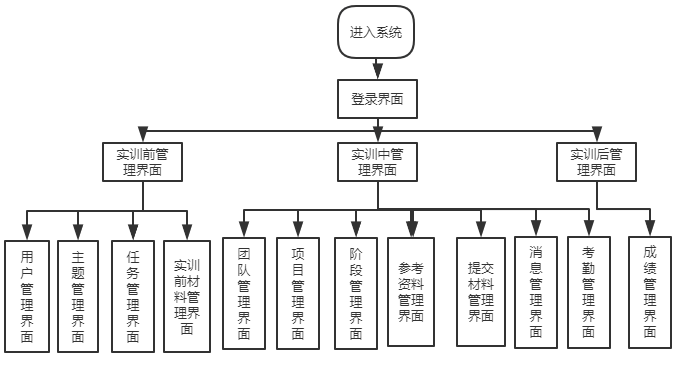


图5-1系统界面关系图

## 界面设计成果

### 主界面

软件校企合作实训平台网站首页，系统管理员、企业教师、学生需输入用户名和密码才能进入系统，首页登录界面第三行是个下拉框，可以选择登录身份。只有学生、老师才需要注册，管理员信息由超级管理员管理。



图5-2 系统登录主界面

管理员端主页面



图5-3 管理员端主界面

教师端主页面



图5-4 教师端主界面

教师端主页面



图5-5 学生端主界面

### 子界面

管理员端的学生管理子页面



图5-6 管理员端的学生管理子页面

管理员端的任务管理子页面



图5-7 管理员端的任务管理子页面

管理员端的信息管理子页面



图5-8 管理员端的信息管理子页面

管理员端的教师统计子页面

图5-9 管理员端的教师统计子页面

教师端的阶段查询子页面



图5-10 教师端的阶段查询子页面

教师端的提交材料查询子页面

图5-11 教师端的提交材料查询子页面

教师端的请假审批查询子页面

图5-12 教师端的请假审批查询子页面

教师端的提交材料统计子页面



图5-13 教师端的提交材料统计子页面

教师端的成绩查询子页面

图5-14 教师端的成绩查询子页面

学生端的请假管理子页面

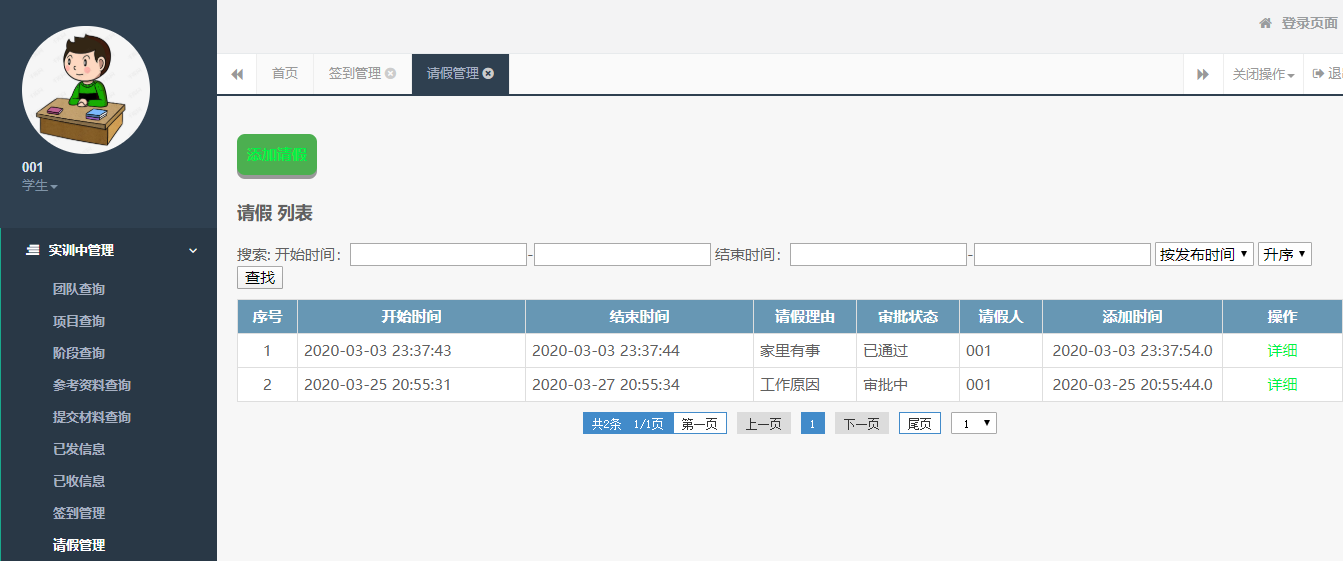


图5-15 学生端的请假管理子页面

学生端的成绩统计子页面



图5-16学生端的成绩统计子页面

6 详细设计

## 系统主要功能模块介绍

1. 用户权限管理主要是系统管理员对不同用户群体设定相应的权限管理，具体功能包括系统管理员注册、登录，系统管理员删除用户、系统管理员给老师、学生用户添加权限和修改权限。
2. 任务管理主要是系统管理员在实训前将所需的实训前准备任务下发给企业教师，具体功能包括浏览、修改任务信息，发布、撤销任务。
3. 项目管理工作主要是由企业教师在项目发布前发布、撤销项目，修改、浏览项目信息。学生可以查看个人项目信息。
4. 阶段管理主要是企业教师对实训期间每个阶段进行宏观把控管理，主要包括阶段基本信息管理、进程控制和阶段评审等，其中阶段基本信息管理包括添加阶段信息、删除阶段信息、删除阶段信息和浏览阶段信息。学生可以查看个人阶段信息。
5. 提交材料管理主要是学生在实训期间，提交上传相关材料给企业老师，企业老师收到学生提交的材料后，可以进行任务材料的下载，也可以进行材料的打回。
6. 考勤管理主要是针对学生实训阶段的考勤管理，主要功能包括按时签到、请假事项、查看个人签到记录、教师查看学生考勤表。
7. 成绩管理主要是在团队提交作品和所有相关材料之后，企业教师对学生进行成绩评定的管理，支持填写学生成绩或修改学生成绩等。学生可以查看自己的成绩。

## 6.2用户管理

6.2.1 功能结构设计

用户管理功能结构图

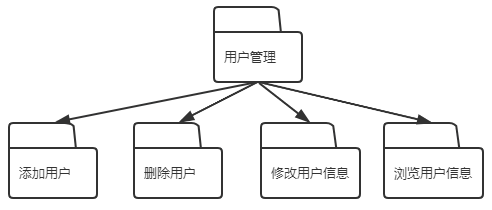


图6-1 用户管理功能结构设计图

用户管理主要是系统管理员对不同用户群体设定相应的权限管理，具体功能包括系统管理员添加、删除用户、系统管理员给老师、学生用户修改信息及浏览用户信息。

6.2.2 类图设计

用户管理类图

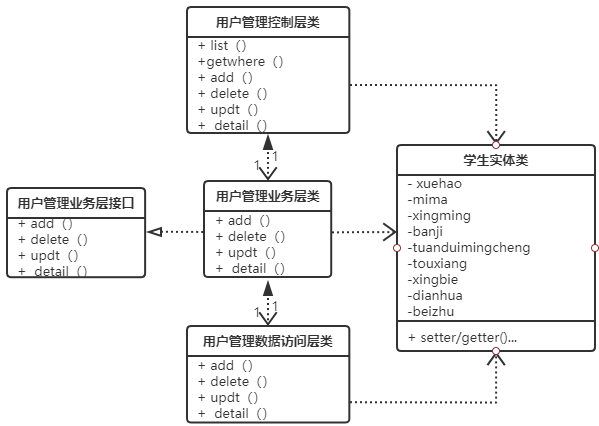


图6-2 用户管理类图

用户管理控制类描述

表6-1 用户管理控制类

|  |  |  |
| --- | --- | --- |
| 用户管理控制类 | | |
| 类的描述：根据前台的业务逻辑的不同跳转到不同方法，经过控制器类再跳转至前台展示 | | |
| 操作描述 | list() | 展示各个端的用户信息列表 |
| getWhere() | 进行sql语句的拼接 |
| add（） | 增加用户 |
| delete（） | 删除用户 |
| updt（） | 更新用户信息 |
| detail（） | 浏览用户信息 |

用户管理业务层类描述

表6-2 用户管理业务层类

|  |  |  |
| --- | --- | --- |
| 用户管理业务层类 | | |
| 类的描述：用户信息管理的实现方法 | | |
| 操作描述 | add（） | 增加用户方法 |
| delete（） | 删除用户方法 |
| updt（） | 更新用户信息方法 |
| detail（） | 浏览用户信息方法 |

用户管理业务层接口类描述

表6-3 用户管理业务层接口类

|  |  |  |
| --- | --- | --- |
| 用户管理业务层接口类 | | |
| 类的描述：用户信息管理的业务层接口类 | | |
| 操作描述 | add（） | 增加用户 |
| delete（） | 删除用户 |
| updt（） | 更新用户信息 |
| detail（） | 浏览用户信息 |

用户管理数据访问层类描述

表6-4 用户管理数据访问层类

|  |  |  |
| --- | --- | --- |
| 用户管理数据访问层类 | | |
| 类的描述：用户信息管理的数据访问层类 | | |
| 操作描述 | add（） | 增加用户 |
| delete（） | 删除用户 |
| updt（） | 更新用户信息 |
| detail（） | 浏览用户信息 |

阶段实体类描述

表6-5 学生实体类

|  |  |  |
| --- | --- | --- |
| 学生实体类 | | |
| 类的描述：学生实体类 | | |
| 属性  描述 | Xuehao | 学号 |
| Mima | 姓名 |
| Banji | 班级 |
| Tuanduimingcheng | 团队名称 |
| Touxiang | 头像 |
| Xingbie | 性别 |
| Dinahua | 电话 |
| Beizhu | 备注 |
| 操作描述 | setter/getter() | 对学生属性的set和get方法 |

6.2.3 顺序图设计

用户管理顺序图

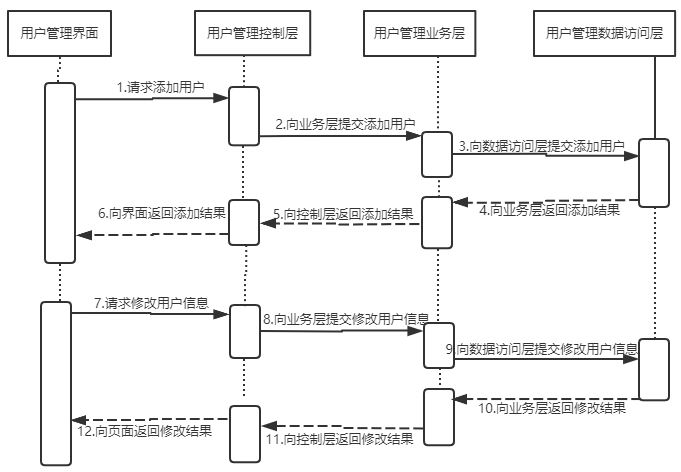


图6-3 用户管理顺序图

6.2.4 核心处理流程设计

查询用户信息活动图

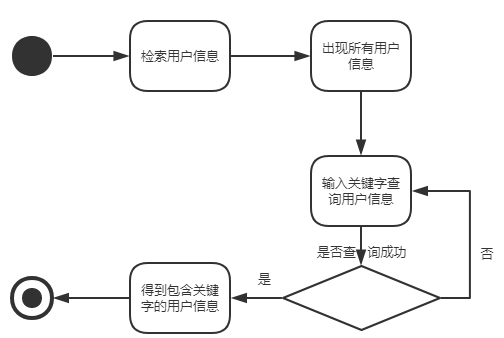


图6-4 查询用户信息活动图

## 6.3项目管理

### 6.3.1功能结构设计

项目管理功能结构设计图

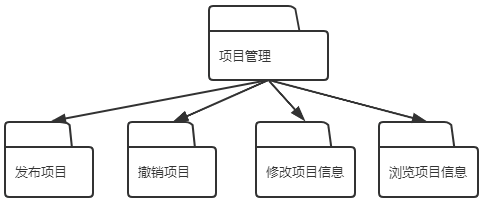


图6-9 项目管理功能结构设计图

项目管理模块是企业教师在实训中对学生项目的统一管理功能模块。由项目启动前浏览查看项目详细信息、修改项目的各项信息、发布项目、撤销项目四个子模块功能组成。学生也可以在实训中查看项目信息。

6.3.2类图设计

项目管理类图

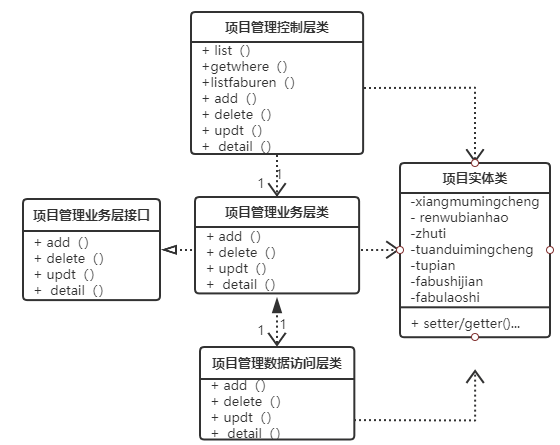


图6-10 项目管理类图

项目管理控制类描述

表6-11 项目管理控制类

|  |  |  |
| --- | --- | --- |
| 项目管理控制类 | | |
| 类的描述：根据前台的业务逻辑的不同跳转到不同方法，经过控制器类再跳转至前台展示 | | |
| 操作描述 | list() | 展示学生端项目信息列表 |
| getWhere() | 进行sql语句的拼接 |
| Listfaburen() | 展示教师端项目信息列表 |
| add（） | 发布项目 |
| delete（） | 撤销项目 |
| updt（） | 更新项目信息 |
| detail（） | 浏览项目信息 |

项目管理业务层类描述

表6-12 项目管理业务层类

|  |
| --- |
| 项目管理业务层类 |
| 类的描述：项目信息管理的实现方法 |

续表6-12

|  |  |  |
| --- | --- | --- |
| 操作描述 | add（） | 发布项目方法 |
| delete（） | 撤销项目方法 |
| updt（） | 更新项目信息方法 |
| detail（） | 浏览项目信息方法 |

项目管理业务层接口类描述：

表6-13 项目管理业务层接口类

|  |  |  |
| --- | --- | --- |
| 项目管理业务层接口类 | | |
| 类的描述：项目信息管理的业务层接口类 | | |
| 操作描述 | add（） | 发布项目 |
| delete（） | 撤销项目 |
| updt（） | 更新项目信息 |
| detail（） | 浏览项目信息 |

项目管理数据访问层类描述：

表6-14 项目管理数据访问层类

|  |  |  |
| --- | --- | --- |
| 项目管理数据访问层类 | | |
| 类的描述：项目信息管理的数据访问层类 | | |
| 操作描述 | add（） | 发布项目 |
| delete（） | 撤销项目 |
| updt（） | 更新项目信息 |
| detail（） | 浏览项目信息 |

项目实体类描述

表6-15 项目实体类

|  |  |  |
| --- | --- | --- |
| 项目实体类 | | |
| 类的描述：项目信息实体类 | | |
| 属性描述 | Renwubianhao | 任务编号 |
| Xiangmumingcheng | 项目名称 |
| Tuanduimingcheng | 团队名称 |
| Zhuti | 主题 |
| Tupian | 图片 |
| Fabushijian | 发布时间 |
| Fabulaoshi | 发布老师 |
| 操作描述 | setter/getter() | 对项目属性的set和get方法 |

### 6.3.3顺序图设计

项目管理顺序图

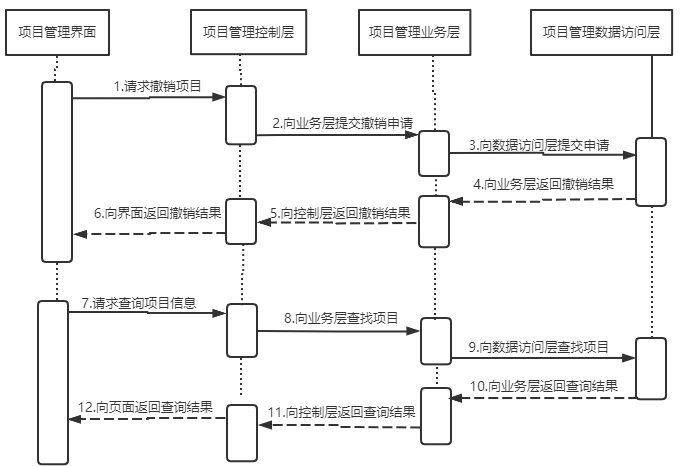


图6-11 项目管理顺序图

### 6.3.4核心处理流程设计

查询项目信息活动图

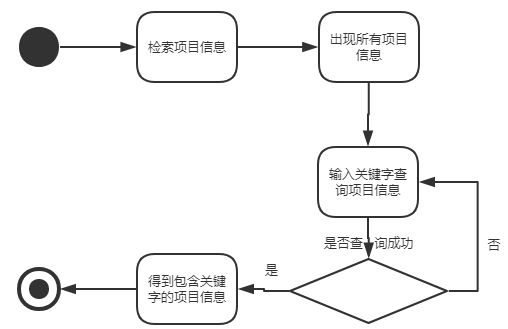


图6-12 查询项目信息活动图

## 6.4阶段管理

### 6.4.1功能结构设计

阶段管理功能架构设计图

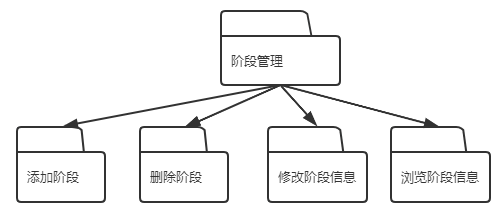


图6-13 阶段管理功能结构设计图

阶段管理主要是在实训项目发布后，企业教师对实训阶段的管理把控。其中包括阶段基本信息管理、阶段评审等功能。学生也可以查看个人阶段信息。

### 6.4.2类图设计

阶段管理类图

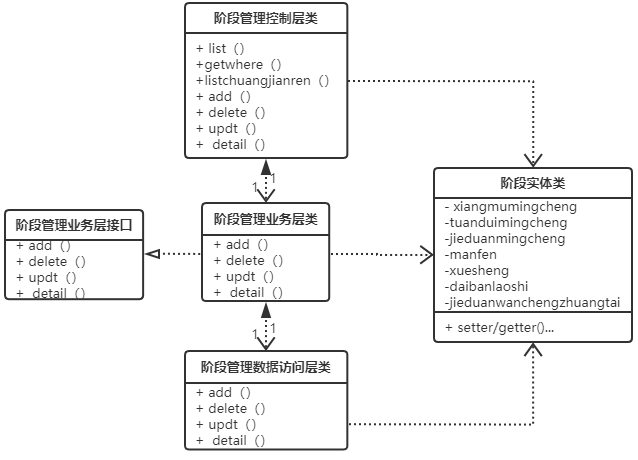


图6-14 阶段管理类图

阶段管理控制类描述

表6-16 阶段管理控制类

|  |  |  |
| --- | --- | --- |
| 阶段管理控制类 | | |
| 类的描述：根据前台的业务逻辑的不同跳转到不同方法，经过控制器类再跳转至前台展示 | | |
| 操作描述 | list() | 展示学生端阶段信息列表 |
| getWhere() | 进行sql语句的拼接 |
| listchuangjianren() | 展示教师端阶段信息列表 |
| add（） | 增加阶段 |
| delete（） | 删除阶段 |
| updt（） | 更新阶段信息 |
| detail（） | 浏览阶段信息 |

阶段管理业务层类描述

表6-17 阶段管理业务层类

|  |
| --- |
| 阶段管理业务层类 |
| 类的描述：阶段信息管理的实现方法 |

续表6-17

|  |  |  |
| --- | --- | --- |
| 操作描述 | add（） | 增加阶段方法 |
| delete（） | 删除阶段方法 |
| updt（） | 更新阶段信息方法 |
| detail（） | 浏览阶段信息方法 |

阶段管理业务层接口类描述

表6-18 阶段管理业务层接口类

|  |  |  |
| --- | --- | --- |
| 阶段管理业务层接口类 | | |
| 类的描述：阶段信息管理的业务层接口类 | | |
| 操作描述 | add（） | 增加阶段 |
| delete（） | 删除阶段 |
| updt（） | 更新阶段信息 |
| detail（） | 浏览阶段信息 |

阶段管理数据访问层类描述

表6-19 阶段管理数据访问层类

|  |  |  |
| --- | --- | --- |
| 阶段管理数据访问层类 | | |
| 类的描述：阶段信息管理的数据访问层类 | | |
| 操作描述 | add（） | 增加阶段 |
| delete（） | 删除阶段 |
| updt（） | 更新阶段信息 |
| detail（） | 浏览阶段信息 |

阶段实体类描述

表 6-20 阶段实体类

|  |  |  |
| --- | --- | --- |
| 阶段实体类 | | |
| 类的描述：阶段信息实体类 | | |
| 操作描述 | xiangmumingcheng | 项目名称 |
| Jieduanmingcheng | 阶段名称 |
| Tuanduimingcheng | 团队名称 |
| Manfen | 满分 |
| Xuesheng | 学生 |
| Daibanlaoshi | 带班老师 |
| Jieduanwanchengzhuangtai | 阶段完成状态 |
| 操作描述 | setter/getter() | 对阶段属性的set和get方法 |

### 6.4.3顺序图设计

阶段管理顺序图

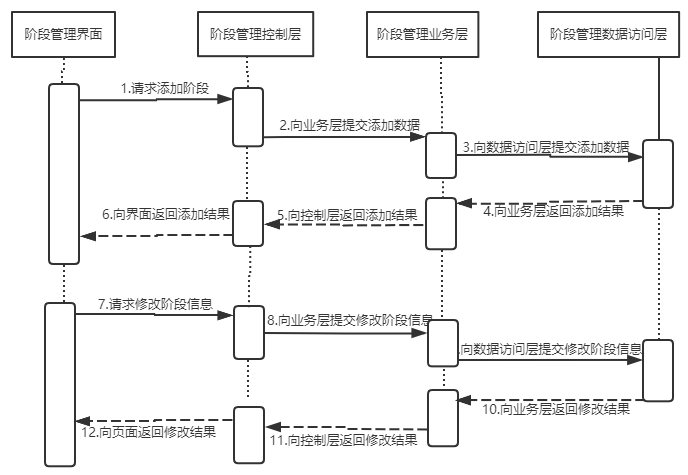


图6-15 阶段管理顺序图

### 6.4.4核心处理流程设计

查询阶段信息活动图

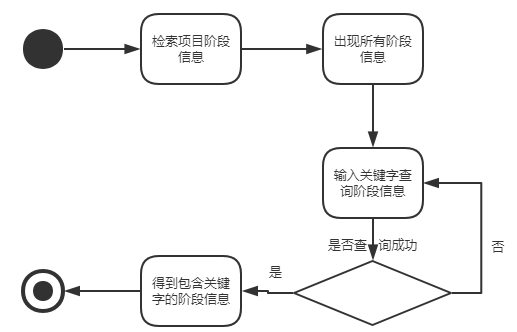


图6-16 查询阶段信息活动图

## 6.5提交材料管理

### 6.5.1功能结构设计

提交材料管理功能架构设计图

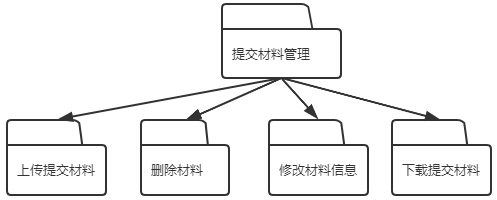


图6-17 提交材料管理功能结构设计图

在实训过程功，老师布置完项目任务后，学生完成项目任务，提交上传相应的文档等材料，老师对其进行审核，如果不通过（例如上传提交材料命名不规范等），将提交材料进行删除。如果审核通过，老师对提交的材料进行下载。

### 6.5.2.类图设计

提交材料管理类图

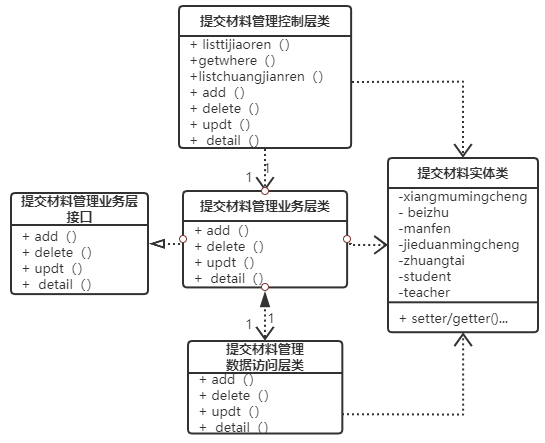


图6-18 提交材料管理类图

提交材料管理控制类描述

表6-21 提交材料管理控制类

|  |  |  |
| --- | --- | --- |
| 提交材料管理控制类 | | |
| 类的描述：根据前台的业务逻辑的不同跳转到不同方法，经过控制器类再跳转至前台展示 | | |
| 操作描述 | Listtijiaoren() | 展示学生端提交材料信息列表 |
| getWhere() | 进行sql语句的拼接 |
| listchuangjianren() | 展示教师端提交材料信息列表 |
| add（） | 上传提交材料 |
| delete（） | 删除材料 |
| updt（） | 修改材料信息 |
| detail（） | 浏览材料信息 |

提交材料管理业务层类描述

表6-22 提交材料管理业务层类

|  |
| --- |
| 提交材料管理业务层类 |
| 类的描述：提交材料信息管理的实现方法 |

续表6-22

|  |  |  |
| --- | --- | --- |
| 操作描述 | add（） | 上传提交材料方法 |
| delete（） | 删除材料方法 |
| updt（） | 修改材料信息方法 |
| detail（） | 浏览材料信息方法 |

提交材料管理业务层接口类描述

表6-23 提交材料管理业务层接口类

|  |  |  |
| --- | --- | --- |
| 提交材料管理业务层接口类 | | |
| 类的描述：提交材料信息管理的业务层接口类 | | |
| 操作描述 | add（） | 上传提交材料 |
| delete（） | 删除材料 |
| updt（） | 修改材料信息 |
| detail（） | 浏览材料信息 |

提交材料管理数据访问层类描述

表6-24 提交材料管理数据访问层类

|  |  |  |
| --- | --- | --- |
| 提交材料管理数据访问层类 | | |
| 类的描述：提交材料信息管理的数据访问层类 | | |
| 操作描述 | add（） | 上传提交材料 |
| delete（） | 删除材料 |
| updt（） | 修改材料信息 |
| detail（） | 浏览材料信息 |

提交材料实体类描述

表6-25 提交材料实体类

|  |  |  |
| --- | --- | --- |
| 提交材料实体类 | | |
| 类的描述：提交材料信息实体类 | | |
| 操作描述 | xiangmumingcheng | 项目名称 |
| Jieduanmingcheng | 阶段名称 |
| Beizhu | 文件名称 |
| Manfen | 满分 |
| Xuesheng | 提交学生 |
| Daibanlaoshi | 审批教师 |
| zhuangtai | 打分状态 |
| setter/getter() | 对提交材料属性的set和get方法 |

### 6.5.3顺序图设计

提交材料管理顺序图

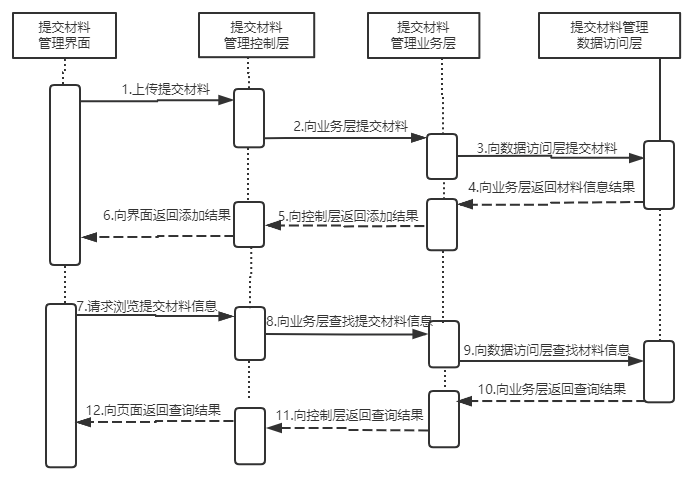


图6-19 提交材料管理顺序图

### 6.5.4核心处理流程设计

删除提交材料活动

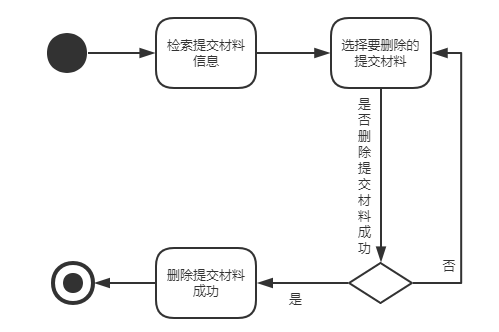


图6-20 删除提交材料活动图

## 6.6考勤管理

### 6.6.1功能结构设计

考勤管理功能架构设计图

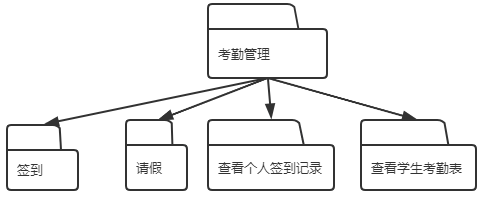


图6-21 考勤管理功能结构设计图

考勤管理模块包括添加人员信息然后学生来负责签到、请假。学生还可以通过此系统查看个人签到记录，企业任课老师可以查看学生的考勤表。

### 6.6.2类图设计

考勤管理类图

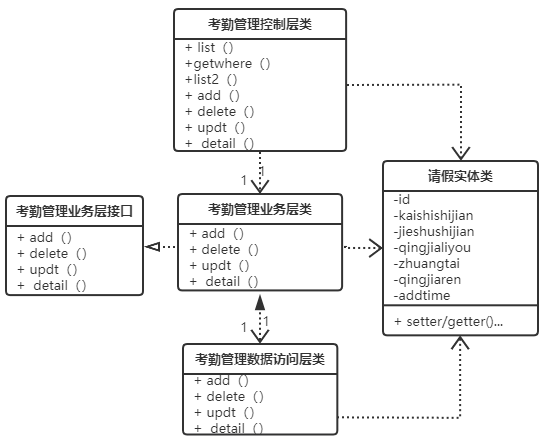


图6-22 考勤管理类图

考勤管理控制类描述

表6-26 考勤管理控制类

|  |  |  |
| --- | --- | --- |
| 考勤管理控制类 | | |
| 类的描述：根据前台的业务逻辑的不同跳转到不同方法，经过控制器类再跳转至前台展示 | | |
| 操作描述 | List() | 展示教师端考勤信息列表 |
| getWhere() | 进行sql语句的拼接 |
| List2() | 展示学生端考勤信息列表 |
| add（） | 签到或请假 |
| delete（） | 删除考勤信息 |
| updt（） | 修改考勤信息 |
| detail（） | 浏览考勤信息 |

考勤管理业务层类描述

表6-27 考勤管理业务层类

|  |
| --- |
| 考勤管理业务层类 |
| 类的描述：考勤信息管理的实现方法 |

续表6-27

|  |  |  |
| --- | --- | --- |
| 操作描述 | add（） | 签到或请假方法 |
| delete（） | 删除考勤信息方法 |
| updt（） | 修改考勤信息方法 |
| detail（） | 浏览考勤信息方法 |

考勤管理业务层接口类描述

表6-28 考勤管理业务层接口类

|  |  |  |
| --- | --- | --- |
| 考勤管理业务层接口类 | | |
| 类的描述：考勤信息管理的业务层接口类 | | |
| 操作描述 | add（） | 签到或请假 |
| delete（） | 删除考勤信息 |
| updt（） | 修改考勤信息 |
| detail（） | 浏览考勤信息 |

考勤管理数据访问层类描述

表6-29 考勤管理数据访问层类

|  |  |  |
| --- | --- | --- |
| 考勤管理数据访问层类 | | |
| 类的描述：考勤信息管理的数据访问层类 | | |
| 操作描述 | add（） | 签到或请假 |
| delete（） | 删除考勤信息 |
| updt（） | 修改考勤信息 |
| detail（） | 浏览考勤信息 |

请假实体类描述

表6-30 请假实体类

|  |  |  |
| --- | --- | --- |
| 请假实体类 | | |
| 类的描述：请假信息实体类 | | |
| 操作描述 | Id | 序号 |
| Kaishishijian | 开始时间 |
| Jieshushijian | 结束时间 |
| Qingjialiyou | 请假理由 |
| Zhuangtai | 审批状态 |
| Qingjiaren | 请假人 |
| Addtime | 添加时间 |
| 操作描述 | setter/getter() | 对请假属性的set和get方法 |

### 6.6.3顺序图设计

考勤管理顺序图

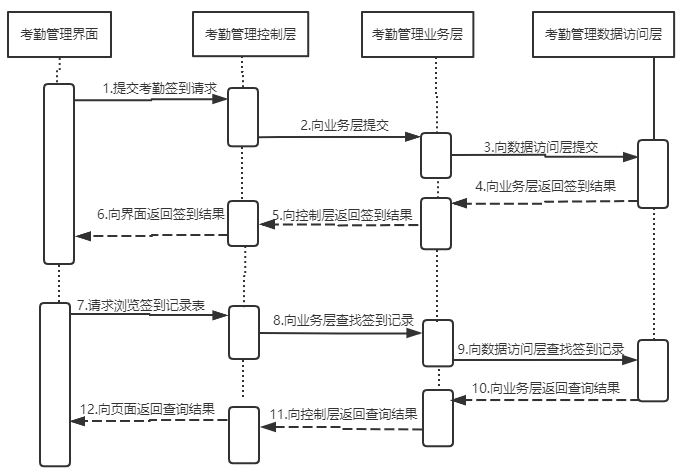


图6-23 考勤管理顺序图

6.6.4核心处理流程设计

请假活动图

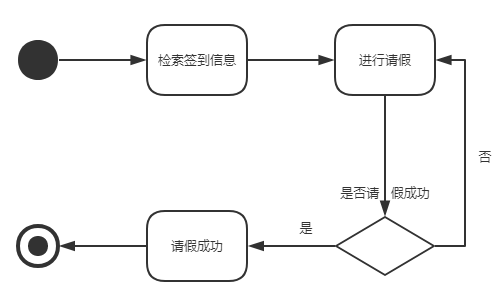


图6-24 请假活动图

## 6.7成绩管理

### 6.7.1功能结构设计

成绩管理功能架构设计图

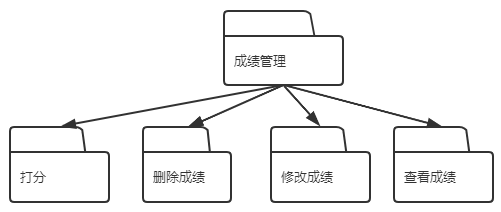


图6-25 成绩管理功能结构设计图

成绩评定开始后，老师根据学生提交的作品和相关材料为学生打分填写成绩，若打分错误可随时更改学生成绩，老师打分后学生可查看自己的成绩。

### 6.7.2类图设计

成绩管理类图

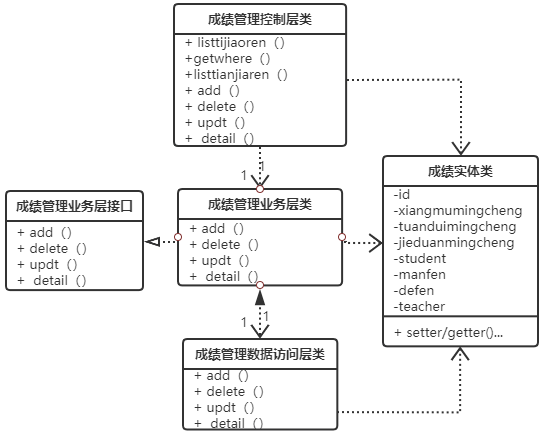


图6-26 成绩评定管理类图

成绩管理控制类描述

表6-31 成绩管理控制类

|  |  |  |
| --- | --- | --- |
| 成绩管理控制类 | | |
| 类的描述：根据前台的业务逻辑的不同跳转到不同方法，经过控制器类再跳转至前台展示 | | |
| 操作描述 | Listtijiaoren() | 展示学生端成绩信息列表 |
| getWhere() | 进行sql语句的拼接 |
| Listtianjiaren() | 展示教师端成绩信息列表 |
| add（） | 打分 |
| delete（） | 删除成绩信息 |
| updt（） | 修改成绩信息 |
| detail（） | 浏览成绩信息 |

成绩管理业务层类描述

表6-32 成绩管理业务层类

|  |
| --- |
| 成绩管理业务层类 |
| 类的描述：成绩信息管理的实现方法 |

续表6-32

|  |  |  |
| --- | --- | --- |
| 操作描述 | add（） | 打分方法 |
| delete（） | 删除成绩信息方法 |
| updt（） | 修改成绩信息方法 |
| detail（） | 浏览成绩信息方法 |

成绩管理业务层接口类描述

表6-33 成绩管理业务层接口类

|  |  |  |
| --- | --- | --- |
| 成绩管理业务层接口类 | | |
| 类的描述：成绩信息管理的业务层接口类 | | |
| 操作描述 | add（） | 打分 |
| delete（） | 删除成绩信息 |
| updt（） | 修改成绩信息 |
| detail（） | 浏览成绩信息 |

成绩管理数据访问层类描述

表6-34 成绩管理数据访问层类

|  |  |  |
| --- | --- | --- |
| 成绩管理数据访问层类 | | |
| 类的描述：成绩信息管理的数据访问层类 | | |
| 操作描述 | add（） | 打分 |
| delete（） | 删除成绩信息 |
| updt（） | 修改成绩信息 |
| detail（） | 浏览成绩信息 |

成绩实体类描述

表 6-35成绩实体类

|  |  |  |
| --- | --- | --- |
| 成绩实体类 | | |
| 类的描述：成绩信息实体类 | | |
| 操作描述 | Id | 序号 |
| XiangmuMingcheng | 项目名称 |
| TuanduiMingcheng | 团队名称 |
| JieduanMingcheng | 阶段名称 |
| Student | 学生 |
| Manfen | 满分 |
| Defen | 得分 |
| Teacher | 打分老师 |
| setter/getter() | 对成绩属性的set和get方法 |

### 6.7.3顺序图设计

成绩管理顺序图

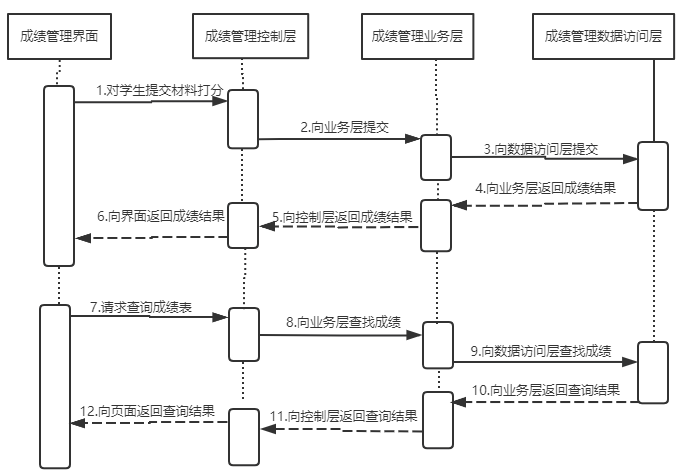


图6-27 成绩评定管理顺序图

6.7.4核心处理流程设计

打分活动图

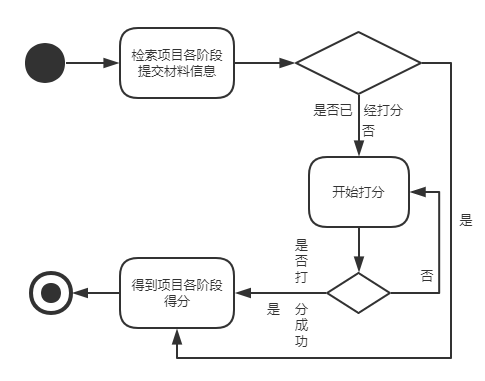


图6-28 打分活动图

7 编码

## 7.1 代码实现与核心算法

（1）实现成绩管理核心代码

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 实现阶段管理核心代码

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 实现请假审批核心代码

|  |
| --- |
|  |

|  |
| --- |
|  |

（4）统计学生各阶段得分核心代码

|  |
| --- |
|  |

（5）实现学生功能相关业务核心代码

|  |
| --- |
|  |

## 7.2 代码优化分析

学生端统计管理中统计项目完成进度的优化

代码优化的目标有：减少代码量，提升程序的速度。由于有多个数据，如带班老师、学生、阶段名称、阶段完成状态。为了减少代码量，减少重复的代码，将两次从阶段表里获取到的数据都存放在List<HashMap>中，一个命名为list，另一个命名为list2。

|  |
| --- |
|  |

8 测试

## 8.1 测试方案设计

### 8.1.1 测试策略

一般来说，实训平台从产品研发到正式发布，以及后期的维护运行中，有较大的概率会遇到严重问题，因此导致较高的系统开发失败的概率。软件校企合作实训平台也是这样的，系统如果出现问题，比如说图片加载出现问题，系统响应太慢，将很大程度影响实训平台的体验，如果不提早解决，等产品上线后很可能会流失大量用户，收到不良口碑从而严重影响市场，同时也将进一步耗费巨大的人力物力来出席进行后期研发与维护，带来极大的损失。

在进行产品测试时，要先验证网站的运行情况是否与预期结果一致。综合测试是功能测试（黑盒测试法）和结构测试（白盒测试法）。黑盒测试是把测试对象当成黑盒，不深究测试对象内部的具体实现，只考虑整体特性，来证明每个实现了的功能是否符满需求；而白盒测试是通过测试对象的内部构造来设计测试用例，用测试用例来测试内部控制流。

### 8.1.2 测试进度安排

本系统的测试进度计划安排如下：

1. 编写制定黑盒测试所需要的测试用例；
2. 用制定好的测试用例进行黑盒测试，并将测试结果总结成报告，同时也要对没有达到预期目的的用例进行记录；
3. 测试系统的健壮性；
4. 完成以上测试之后，通过白盒测试对关键代码和算法进行测试；
5. 提交测试报告。

### 8.1.3 测试资源

（1）测试环境

Windows 10操作系统，1G独立显卡，CPU为INTEL CORE i5处理器，固态硬盘256G，内存8G， IDEA2018，Tomcat8.0.411。

（2）测试管理工具

Bugzila工具。Bugzila是一种轻量级的测试管理工具，用于检测程序是否能够正常运行以及达到预期的功能。Bugzila可以快速地找到控件，通过引用Bugzila.Core下的类的方法和属性的使用，就可以实现功能自动化测试。如今，Bugzilla不光在缺陷管理领域中持续优化产品，还逐步完善了测试管理的其他领域，如测试用例管理、测试流程管理、产品需求管理等，实现了测试的一体化管理。

（3）测试文档

软件测试计划文档、软件测试用例、缺陷报告文档。

### 8.1.4 关键测试点

（1）管理员登录后，进入软件校企合作实训平台。再进入学生管理界面后，是否会出现学生所有信息包括头像，并且点击详情可进入学生信息详情页。

（2）管理员登录后，进入统计管理页面，是否能同时看到学生、教师统计页面，进入各自的详情页后，是否能看到用竖状的条形统计图统计了各自的数量。

（3）管理员登录进入主页面后，再进入消息管理页面，查看是否具有最高权限能看到三种用户管理员、教师、学生之间互相发送的消息。

（4）教师登录后，进入实训中管理页面，找到阶段管理，看是否会出现对应的所有阶段信息，且自己是否能对阶段信息进行基本的操作。

（5）教师登录后，在实训中管理中找到提交材料管理，看是否会出现学生提交的所有材料，且看自己是否能对各个提交材料对应的阶段进行打分操作。

（6）教师登录后，在实训中管理中找到请假管理，看是否能看到自己学生的请假申请，再进行审批，看是否能对审批状态进行修改。

（7）教师登录后，进入统计管理中，再进入提交材料统计页面中，看是否能看到所带学生提交材料的数量，数量将以条形统计图的方式显示。

（8）学生登录后，进入统计管理中，在进入成绩管理页面，看是否会显示自己的各阶段信息，以及各阶段所获得的分数，分值是否会以扇形统计图的比例形式显示出来。

## 8.2 测试用例构建

### 8.2.1 测试用例编写约定

（1）测试用例与开发的对应约定

表 8-1 测试用例与开发约定

|  |  |  |
| --- | --- | --- |
| 开发阶段 | 依据文档 | 编写的用例 |
| 需求分析阶段 | 需求文档 | 对应系统测试的用例 |
| 概要设计阶段 | 概要设计文档 | 对应集成测试的用例 |
| 详细设计阶段 | 详细设计文档 | 对应单元测试的用例 |

1. 测试用例类型约定

表 8-2 测试用例类型约定

|  |  |  |  |
| --- | --- | --- | --- |
| 测试用例 | 对应测试类型 | 测试覆盖率（测试人员） | 测试覆盖率（其他人员） |
| 功能测试用例 | 主要包括实训平台运行过程的可靠性测试、健壮性测试以及功能测试 | 85%~100% | 80% |

续表 8-2

|  |  |  |  |
| --- | --- | --- | --- |
| 管理以及统计界面测试用例 | 个人信息主页测试、项目、阶段、资料查询、成绩查询界面测试以及少量功能测试 | 100% | 85% |

1. 在测试阶段中，类型与执行角色的关系约定

表 8-3 测试阶段类型与执行角色的约定

|  |  |  |
| --- | --- | --- |
| 测试阶段 | 测试类型 | 测试角色 |
| 单元测试 | 模块功能测试 | 主要是开发人员，测试人员可以进行测试配合 |
| 集成测试 | 功能测试（接口、路径测试） | 测试人员 |
| 系统测试 | 功能测试（健壮性、性能、个人信息主页界面、安全性、压力、可靠性等测试） | 测试人员 |

### 8.2.2 测试用例设计

本系统测试用例设计大致

表 8-4 测试用例设计

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 测试项 | 描述/输入/操作 | 期望结果 |
| 01 | 主题查询 | 输入主题名称后，点击搜索 | 展示出对应的主题列表 |
| 02 | 项目查询 | 输入项目名称后，点击搜索 | 在原页面上出现对应的项目列表 |
| 03 | 团队添加 | 输入团队名称等相关信息后，点击添加 | 在团队添加页面上出现新添加的团队信息 |
| 04 | 阶段查询 | 输入阶段名称后，点击搜索 | 在原页面上出现对应的阶段列表 |
| 05 | 上传提交材料 | 学生在阶段查询界面，对应的阶段上点击上传提交材料 | 在提交材料页面出现新上传的提交材料 |
| 06 | 打分 | 教师在提交材料查询界面，下载学生上传的提交材料，查看后，对相应阶段进行输入成绩，打分 | 在成绩页面，出现对应阶段的得分 |

### 8.2.3 关键测试用例

（1）上传提交材料测试

学生可在登录校企合作实训平台后，进入提交材料管理界面，完成老师布置的项目任务后，即可上传提交材料，提交材料能否成功上传，是否会显示通知上传成功，上传之后能否看到材料信息都很重要，上传提交材料测试

表8-5 界面测试结果

|  |  |
| --- | --- |
| Windows 10下IDEA 2018 | 出现结果和预计结果是否一致（是/否） |
| 提交材料能否成功上传 | 是 |
| 能否会显示通知上传成功 | 是 |
| 查询各类信息时实训平台的反应时间能否小于2秒？ | 是 |
| 上传之后能否看到材料信息 | 是 |

（2）打分测试

教师可在登录校企合作实训平台后，进入提交材料管理界面，下载学生上传的提交材料，查看后，对相应阶段进行输入成绩，进行打分操作。能否成功下载学生上传的提交材料，下载后能否查看提交材料，提交材料能否正常显示不乱码，打分能否成功显示成绩都很重要，打分测试

表8-6 界面测试结果

|  |  |
| --- | --- |
| Windows 10下IDEA 2018 | 出现结果和预计结果是否一致（是/否） |
| 能否成功下载学生上传的提交材料 | 是 |
| 下载后能否查看提交材料 | 是 |
| 提交材料能否正常显示不乱码 | 是 |
| 打分能否成功显示成绩 | 是 |

（3）界面测试

软件校企合作实训平台的整体布局与排版对用户的体验至关重要，因此进入实训平台后，需要对查询及管理过程中的界面进行实时观察，如图片是否显示清晰，平台的图文排版是合理，选择展示的统计图是否是最能呈现出实际准确的数量及比例信息等等，界面测试

表8-7 界面测试结果

|  |  |
| --- | --- |
| Windows 10下IDEA 2018 | 是否通过测试（是√否×） |
| 界面的布局是否合理？ | √ |
| 查询详情页的信息排版是否合理？ | √ |
| 查询各类信息时实训平台的反应时间是否小于2秒？ | √ |
| 点击修改打分状态后是否显示为已打分？ | √ |
| 学生教师管理员的头像图片是否清晰？ | √ |

（4）数据显示测试

数据显示测试主要是为了验证在使用校企合作实训平台的过程中，用户的个人页面和各类数据相关功能显示是否实时、准确，其中数据显示测试

表8-7 数据显示测试

|  |  |
| --- | --- |
| Windows 10下IDEA 2018 | 是否通过测试（是√否×） |
| 阶段各项列表显示 | √ |
| 发送消息的时间显示 | √ |
| 学生、老师的数量显示 | √ |
| 提交材料的数量显示 | √ |
| 点击修改状态变为已打分或已通过显示 | √ |

### 8.2.4 测试用例维护

为了满足用户对于实训平台更大范围使用的要求，将要添加、删除和修改部分实训平台功能，于是需要修改、增加部分测试用例来适应实训平台功能，主要的是更改实训平台测试用例的工作有下面几点：

（1）及时删除冗余的测试用例

由于测试用例是由多名测试人员编写，所以存在多个测试用例是测试同一个功能的，这样就会大大降低测试效率，因此需要及时将冗余的用例进行删除，并对用例集进行定期的整理。

（2）增加新的测试用例

从使用实训平台的用户那里得到反馈，收集并统计本实训平台还有哪些不足的地方，还有哪些功能需要改进，需要增加哪些功能，这就需要增加新的测试用例，

（3）及时删除过时的测试用例

随着实训平台功能的更新，部分实训平台功能不再适应新版本功能而被删除，那么用来测试该功能的测试用例也失去了价值，因此就需要将这些无用的用例删除。因此，随着新版本的推出，就需要将无用的测试用例删除。

（4）改进不受控制的测试用例

由于实训平台的功能不断更新，测试人员写的测试用例也越来越多，其中就有部分用例对输入非常敏感，所以很难控制这些用例的结果，因此需要对它们进行改进，让这些用例达到可控制。当然，测试用例的维护是一个改进和完善的过程,同时也是非常长期的过程。

9 总结与展望

## 9.1 设计工作总结

此次毕业设计是做一个软件校企合作实训平台，本来这就是个比较大的系统，涵盖的管理功能比较多比较杂，比我预期的难度要更大，因为整个大学期间都没有独立开发一个完整系统的经历，所以在开发实训平台过程中出现了许多问题，例如头像图片的存储及文件的上传下载等问题。难度比较大的地方是在于数据库的设计和多种多样图表的统计功能，数据库的设计由于要用到很多个表，然后表又有很多数据，表与表之间的数据要设置好外键、主键，彼此之间相关联，要做到在某个模块修改了某张表的某个数据，另外所有与这张表相关地方的数据都要被修改，也就是一改全改功能。

至于多种多样的图表统计功能，则是用到了功能强大的highcharts插件，实训平台里的条形统计图以及扇形统计图都是用这个插件自动生成的，再就是用到了Hashmap来获取从数据库中的值，进行计数、封装，再把它get出来，放到前台显示的界面中。当然也有很多不完善的地方，例如教师端的项目阶段统计只能统计所有学生的完成进度一起统计，并没有做到人性化的每位同学都有一个扇形统计图。

## 9.2 未来工作展望

本实训平台虽然在整体上面有一个基本功能的应用，但是对于一个更完整的实训平台来说还是远远不够的。随着时代的进步，学生、老师对实训情况的了解要求越来越高，但本系统的设计远远不够，因此在后续的开发中，可以在以下几个方面进行改进：

（1）学生实训进度展示方面。利用扇形统计图统计学生的实训项目的阶段，显示出学生项目完成了多少，完成到了哪个阶段。教师要能查看到其所带学生的项目完成进度。

（2）实训平台的信息展现形式。目前展现形式太过单一，几乎都是通过列表的方式展现，往后要多样化、美观化地展现出实训平台的各类信息。

（3）实训平台的页面前端外观效果。现在使用的js动态展示形式较少，将在未来多使用一些，尽可能炫酷地呈现页面。

谢 辞

参考文献

1. Raymond C.Z. Cohen,Simon M. Harrison,Paul W. Cleary. Dive Mechanic: Bringing 3D virtual experimentation using biomechanical modelling to elite level diving with the Workspace workflow engine[J]. Mathematics and Computers in Simulation,2020,175.
2. Erica Mourão,João Felipe Pimentel,Leonardo Murta,Marcos Kalinowski,Emilia Mendes,Claes Wohlin. On the performance of hybrid search strategies for systematic literature reviews in software engineering[J]. Information and Software Technology,2020,123.
3. Luigi Ranghetti,Mirco Boschetti,Francesco Nutini,Lorenzo Busetto. “sen2r”: An R toolbox for automatically downloading and preprocessing Sentinel-2 satellite data[J]. Computers and Geosciences,2020,139.

[4]Patrizio Pelliccione,Eric Knauss,S. Magnus Ågren,Rogardt Heldal,Carl Bergenhem,Alexey Vinel,Oliver Brunnegård. Beyond connected cars: A systems of systems perspective[J]. Science of Computer Programming,2020,191.

[5]Jagdeep Singh,Sachin Bagga,Ranjodh Kaur. Software-based Prediction of Liver Disease with Feature Selection and Classification Techniques[J]. Procedia Computer Science,2020,167.

[6]Christian Lettner,Michael Moser,Josef Pichler. An integrated approach for power transformer modeling and manufacturing[J]. Procedia Manufacturing,2020,42.

[7]Eduardo Juárez, Rocio Aldeco-Pérez, Jose.Manuel Velázquez. Academic approach to transform organisations: one engineer at a time. 2020, 14(2):106-114.

[8]Guadalupe-Isaura Trujillo-Tzanahua, Ulises Juárez-Martínez, Alberto-Alfonso Aguilar-Lasserre, et al. Multiple software product lines to configure applications of internet of things. 2020, 14(2):165-175.

[9]Christopher Theisen,Laurie Williams. Better together: Comparing vulnerability prediction models[J]. Information and Software Technology,2020,119.

[10]Ryan C. Olley,Sana Alhaij,Basim M. Mohsen,Paul L. Appleton,R. Graham Chadwick,Graeme Ball. Novel Confocal-Laser-Scanning-Microscopy and conventional measures investigating eroded dentine following dentifrice dab-on and brushing abrasion[J]. Heliyon,2020,6(3).

[11] (美)阿诺德，Ken Arnold，等.Java程序设计语言[M].北京：人民邮电出版社，2018.

[12]（美）Bruce Eckel著，陈昊鹏译，Java编程思想（第三版）［Ｍ］．北京：机械工业出版社，2017.6.

[13]张宏升. “JavaEE软件技术”课程的“2+2”创新教学模式研究[J]. 现代信息科技,2017.

[14]唐权. SSM框架在JavaEE教学中的应用与实践[J].福建电脑2017.

[15]李杉,贾彦平,达虎. Mybatis逆向工程在JavaEE中的应用[J]. 通讯世界,2017

[16]乔岚. 基于MyBatis和Spring的JavaEE数据持久层的研究与应用[J]. 信息与电脑(理论版),2017.

[17]傅莞龙,张传武,彭安金. 使用Spring Data和JPA在JavaEE系统中简化持久层[J]. 电子世界,2017.

[18]林萍,朱婵. 基于Ajax技术和JAVAEE的分页查询优化[J]. 计算机系统应用,2017.

[19]李瑞,徐家喜,卢迪,周涛春. 基于JavaEE组合测试系统的设计与实现[J]. 电脑知识与技术,2018.

[20]于亚芳,郭磊. JavaEE框架技术课程建设研究[J]. 电脑知识与技术,2018.

[21]陈丽丽. 互联网时代JavaEE计算机软件设计与开发技术研究[J]. 中国新通信,2017.

[22]艾灵仙. 高校系级网站构建平台的设计与实现.[J]. 科技信息-2018年16期.

[23]张超. 基于 JSP 的数据库连接技术浅析[J]. 福建电脑, 2018.

[24]陈倩, 刘胜, 焦壵. 基于 JSP 的培训管理信息系统研究[J]. 机械, 2018.

[25]王莎莎. 基于 JSP 的在线考试系统设计与实现[J]. 计算机光盘软件与应用, 2018.

[26]李宏利,李梅,鱼晓. JavaEE技术课程资源建设方法研究[J]. 黑龙江教育(高教研究与评估),2018.

[27]郝平. 基于JavaEE的学生社团管理系统的设计与实现[J]. 信息与电脑(理论版),2018.

[28]韦佳佳,任海鹏,孙宇. JavaEE在轻量级智慧校园架构设计中的应用[J].太原学院学报(自然科学版),2018.

[29]陈佳,简岩,杨道平. 基于JavaEE的桐城交友网站的设计与实现[J]. 科技经济导刊,2018,.

[30]强,郭标,吴艳文. 触发器在JavaEE开发中应用的研究[J]. 合肥师范学院学报,2018.

附录A 外文翻译—原文部分

**Source:《Java Web integration development and project practice: JSP Ajax, Struts, Hibernate Spring(version 2)》**

**JavaScript introduction**

The precursor of JavaSeript was LiveScript. Since the introduction of the most popular Java language by Sun, Netscape has introduced the concept of Java language programs, making it a "powerful, secure, object - and event-driven language.

1. Advantages of JavaScript

(1) simplicity. It USES small program fragments to embed HTML to achieve programming. Like other scripting languages, JavaScript is an interpreted language that provides a simple development process. Its basic structural form and C, VB, Delphi is very similar. But instead of having to be compiled first, as in these languages, it is explained line by line as the program runs and is combined with HTML tags to make it easier for the user to use.

(2) dynamic. It can respond directly to user input without going through the Web service program. The response to the user is done in an event-driven manner. The so-called event-driven, refers to the home page to perform a certain action produced by the action, called "events." For example, pressing a mouse, moving a window, selecting a menu, and so on can all be considered events. When an event occurs, it may cause a corresponding event response, that is, event handling.

(3) cross-platform. JavaScript is an object-based language that is dependent on the browser itself, independent of the operating environment, and can be executed correctly by any computer that supports a JavaScript browser.

(4) save the interaction time of CGI. With the rapid development of WWW, there are many services provided by the WWW server to communicate with the viewer, to determine the identity of the viewer, the content of the required services, etc., this work is usually written by CGI corresponding procedures and user interaction to complete. Obviously, the interaction between the network and the user increases the traffic of the network on the one hand, and affects the service performance of the server on the other hand. When a server runs a CGI for a user, it needs a process to serve it, which takes up the server's resources (such as CPU service, memory consumption, etc.). If the user fills in the form and there is an error, the time taken up by the interactive network service will increase accordingly. The more the hotspot host being accessed interacts with the user, the greater the performance impact on the server and the slower the network. But not the number of JavaScript, when users fill out a form in the browsing, validation, the interaction of the process just by measuring the browser to call the JavaScript source code in the HTML document explaining to perform to complete, even if you must call the part of CGI, the browser will only after the user input validation information to a remote server, greatly reduced the cost of the server.

2. The working process of JavaScript

Server side

(1) browser input URL request.

(2) the server invokes the script, generates objects that pass data from the browser, and provides these objects to the script.

(3) the script processes (specific database operations are required for database applications) and returns the data through the server in the form of an HTML file to the browser that made the request.

The client

The most typical application of JavaScript is developing client-side scripts.

(1) the JavaScript script is embedded in the HTML file and downloaded to the browser side with the HTML file.

(2) the browser reads the HTML file, then interprets the execution and displays the elements.

(3) if parsing (reading an HTML file and identifying the elements within it is called parsing) comes to a JavaScript script, the browser executes its script statements to do so.

3. Differences between JavaScript and Java

There is a connection between JavaScript and Java, but it is not as close as you might think, and the differences between the two are as follows.

(1) they are two different products developed by two companies. Java, a new object-oriented programming language from Sun, is particularly suited to Intermet application development: JavaScript, a Netscape product designed to extend the capabilities of the Netscape Navigator browser, is an object - and event-driven, interpreted language that can be embedded in Web pages.

(2) JavaScript is object based, while Java is object oriented. Java is a true object-oriented language, and even developing simple programs requires designing objects. JavaScript is a scripting language that can be used to make complex software that has nothing to do with the web and only interacts with users. It is an object - and event-driven programming language, so it provides a very rich set of internal objects for designers to use.

(3) the two languages execute differently in their browsers. Java source code must be compiled before it can be passed to the client for execution, so the client must have an emulator or interpreter on the corresponding platform, which can be used to implement the binding of compiling code independent of a particular platform. JavaScript is an interpretive programming language in which the source code is not compiled before being sent to the client for execution. Instead, the source code is sent to the client in alphabetical format for execution by the browser.

(4) the variables adopted by the two languages are different. Java USES strongly typed variable checking. Variable declaration in JavaScript, with its weak type. That is, variables are not declared before they are used, and the interpreter checks their data types at run time

(5) the code format is not the same. Java is an html-independent format that must be loaded as if it were an external medium referenced in HTML, and its code is stored in a separate document as bytecode. JavaScript code is a text character format that can be embedded directly in HTML documents and dynamically loaded. Writing HTML documents is like editing a text file.

(6) different embedding methods. In HTML documents, the two programming languages have different identities, JavaScript USES is used for identification, while Java USES <applet>.. </applet> to identify.

(7) static binding and dynamic binding. Java is statically linked, meaning that Java's object references must be made at compile time to enable the compiler to implement strong type checking. JavaSeript USES dynamic linking, in which object references to JavaScrpit are checked at run time and cannot be checked for object references without compilation.

4. The use of JavaScript in web pages

There are two ways to add JavaScript to a web page.

(1) directly add HTML documents. This is the most common method used by most web pages with JavaSeript.<script>...</script>. but<script language-"Javascript" >It is used to tell the browser that this is a program written in JavaScript and needs to be interpreted by the corresponding interpreter. The HTML comment tag <1--> is used to remove JavaScript source code that is not recognized by the browser, which is useful for browsers that do not support the JavaSeript language.

//- end of Javascript: double slashes indicate that the comment part of Javascript, that is, characters from the beginning of/to the end of the line, are ignored. The document.write0 function used in the program means that the text in parentheses is output to the window, which will be described in more detail later. Also, it is important to note that < SCP... /scrip> is not fixed and can be included in ... The < / head > > or < body >... anywhere.

1. reference method. If a JavaScript source file (with a js extension) already exists, you can use this reference to increase the utilization of your program code. The url is the address of the program file. Again, such statements can be placed in any part of the HTMIL document header or body. To achieve the effect in the direct insert example, you can first create a JavaSeript source code file called "Scriptjs".

**Summary of the Servlet**

Servlets, one of Sumn's earliest java-based technologies for Web development, have been around for a long time, and Sun has included them in the latest Java ee apis. For starters, it is necessary to know something about servlets, and this chapter looks at the use of servlets from a practical perspective.

Servlets are server-side application components based on Java technology. It's been around for a long time, prior to the J2EE platform. Servlets have been widely used in the past and are still widely used in J2EE project development. Now that servlets are a very mature technology, you can see in the following sections that combining JSPS with servlets is effective. It can completely replace cGl programs in the Web application architecture. In comparison, Serlet has the following advantages.

High execution efficiency: each Servlet corresponds to a separate "thread", so the execution efficiency is higher and more stable than cGI programs.

Low barriers to entry: servlets are java-based, and programmers familiar with Java can easily write a wide variety of Servlet components.

Cross-platform execution capability: Java's own cross-platform capability servlets allow you to run on any java-enabled platform. For servlet-related classes, see the JavaEE API documentation provided by Sun, which provides a set of standards for servlets.

Sun only defines a series of standards, and different vendors can provide their own implementation of the standard, and then choose its specific implementation as needed. Here we use a concrete implementation of the Servlet interface provided by tomet. These implementation classes are stored in the srvet-apijar under the lib folder in the tomeat directory. Interested readers can unzip it and compare it to the content in the API documentation.

1. Servlet life cycle

(1) load the Servlet class. This can happen when the container is started or when the first customer USES it.

(2) instantiation. When the Servlet container receives the first pin request for this Servlet, it invokes the Servlet's default constructor to create an instance of the Servlet.

(3) initialization. When a Servlet instance is created, the container invokes the Servlet's init0 method (which is called only once for a Servlet class).

(4) services. If the Servlet is requested, the container starts a new thread and calls the Servlet's service0 method, which determines why the HTTP request is made, and then calls the corresponding doGet(or doPotO) method.

(5). When the Servlet container determines that a Serlet instance is no longer needed, the destroy0 method is called before the instance is destroyed. (6) not available. Destroy the instance and mark it for garbage collection.

2.three important moments in your life cycle

(1) the initialization

After the Servlet instance is created, and before the Servlet serves a customer request, the container creates a ServletConfig object for the Servlet and invokes the init0 method.

The ServletConfig object is also called the Servlet start parameter object, which has the following meaning.

When a Servlet instance is created, the container creates a ServletConfig object for the Servlet before the Servlet can serve a customer request. Each Servlet has a ServletConfig object. Use to pass deployment time information (Servlet initialization parameters) to the Servlet without having to hard-code this information into the Servlet. Used to access the servletcontext.parameter is configured in the deployment description file.

A ServletContext object, also called a Web application start parameter object, has the following meaning.

When the Servlet container initializes a Web application, it creates a ServletContext object for that application. Each Web application has only one ServletContext object. The starting parameter used to access the Web application. Used to get server information, including the name and version of the container, the version of the supported API, and so on.

(2) service

When the first - client request arrives, the container starts - a new thread, calling the srice0 method, and the srvice0 method calls doGet0 or doPost0 based on the HTTP method (GET, POST, etc.) of the request.

(3) the destruction

If the Servlet container determines that a Servlet entity is no longer necessary or that the Sevler container itself must be shut down, the Servlet container invokes the Servlet's dstor0 method. Free the memory space and associated resources occupied by this Serlet instance.

**JSP overview**

JSP (Java Server Pages) is a dynamic web technology standard initiated by Sun Microsytems. JSP technology USES the Java language as a scripting language. JSP web pages provide an interface for the entire server-side Java library unit to serve HTTP applications.

JSP(Java Server Pages) has its powerful and special function in the construction of dynamic web Pages. JSP is very similar to Microsoft's ASP technology. Both provide the ability to mix some program code in HTML code, with the language engine interpreting and executing the program code. Let's briefly introduce it. JSP pages are eventually converted into servlets. So, basically, anything a JSP page can do can be done with servlets. However, this underlying equality does not mean that servlets and JSP pages are equally applicable in all cases. The problem is not the capability of the technology, but the difference in convenience, productivity, and maintainability. After all, what can be done in the Java programming language on a particular platform can be done in assembly, but it's still important to choose which language.

Add Java program fragments (scriptlets) and JSP tags (ag) to the traditional web page HIML file (.htm," htma) to form a JSP page (\* JSP). When the Web server encounters a request to access a JSP page, it first executes the program fragment and then returns the execution result to the customer in HTML format. Snippets can manipulate databases, redirect web pages, send e-mails, and so on, which is what you need to build a dynamic web site. All the program operations are performed on the server side, the network upload to the client is only the result. Minimum client browser requirements: plugin.activex, Java applets, or even Fame. Compared to using servlets alone, JSPS provide the following benefits:

HTML in JSP is easier to write and maintain. Regular HTML can be used in JSPS: no extra backslashes, no extra double quotes, and no underlying Java syntax. Ability to use standard web development tools. Even HTML tools that know nothing about JSPS can be used because they ignore JSP tags.

The advantages of JSP are as follows:

Updates to the user interface are actually made by the Web Server, so it feels like they are updated quickly. All applications are server-based, so they can be kept up to date. The client interface is not cumbersome and is easy to deploy, maintain, and modify for a variety of applications.

附录B 外文翻译—译文部分

来源：《Java Web整合开发与项目实战：JSP Ajax Struts Hibernate Spring(第2版)》

**JavaScript简介**

JavaSeript语言的前身名叫LiveScript。自Sun公司推出目前最流行的Java语言之后，Netscape公司便将Java语言程序的有关概念引入其中，造就了它是一“种基于对象和事件驱动并具有安全性能的功能强大的语言。

1.JavaScript具有的优点

(1)简单性。它采用小程序片段的方式嵌入HTML来实现编程。像其他脚本语言一样，JavaScript同样是-种解释性语言，它提供了一个简易的开发过程。它的基本结构形式与C、VB、Delphi十分类似。但它不像这些语言一样，需要先编译，而是在程序运行过程中被逐行地解释，并与HTML标识结合在-一起，从而方便用户的使用操作。

(2)动态性。它可以直接对用户的输入做出响应，无须经过Web服务程序。对用户的响应，是采用以事件驱动的方式进行的。所谓事件驱动，就是指在主页中执行了某种操作所产生的动作，就称为“事件”。例如，按下鼠标、移动窗口、选择菜单等都可以视为事件。当事件发生后，可能会引起相应的事件响应，即事件处理。

(3)跨平台性。JavaScript 是基于对象的语言，依赖于浏览器本身，与操作环境无关，只要能支持运行JavaScript浏览器的计算机都可以正确执行。

(4)节省CGI的交互时间。随着www的迅速发展，有许多www服务器提供的服务要与浏览者进行交流，确定浏览者的身份、所需服务的内容等，这项工作通常由CGI编写相应的程序与用户进行交互来完成。很显然，通过网络与用户的交互过程一方面增大了网络的通信量，另一方面影响了服务器的服务性能。服务器为一个用户运行一个CGI时，需要一个进程为它服务，它要占用服务器的资源(如CPU服务、内存耗费等)，如果用户填表出现错误，交互网络服务占用的时间就会相应增加。被访问的热点主机与用户交互越多，服务器的性能影响就越大，网络越慢。但JavaScript则不然，当用户在浏览中填表、验证的交互过程只是通过测览器对调入HTML文档中的JavaScript源代码进行解释执行来完成的，即使是必须调用CGI的部分，浏览器只将用户输入验证后的信息提交给远程的服务器，大大减少了服务器的开销。

2 .JavaScript 的工作过程

服务器端

(1)浏览器输入URL请求。

(2)服务器调用脚本，生成从浏览器传递数据的对象，并向脚本提供这些对象。

(3)脚本进行处理(对于数据库应用则需要进行特定的数据库操作)，并将数据以HTML文件的方式通过服务器返回发出请求的浏览器。

客户端

JavaScript最典型的应用就是开发客户端脚本。

1. JavaScript脚本程序被嵌入到HTML文件中，随着HTML文件一同下载到浏览器端。
2. 浏览器读HTML文件，然后解释执行并显示其中的元素。

(3)如果在语法分析或解析(parsing) (读取HTML文件并分辨其中的元素称为解析)到JavaScript脚本，浏览器就去执行其脚本语句做相应的处理。

3. JavaScript 与Java之间的区别

JavaScript与Java之间存在着联系，但它们之间的联系并不像想象中的那样紧密，二者的区别体现在以下方面。

(1)它们是两个公司开发的不同的两个产品。Java 是Sun公司推出的新- -代面向对象的程序设计语言，特别适合Intermet 应用程序开发:而JavaScript是Netscape 公司的产品，其目的是为了扩展Netscape Navigator浏览器功能，是一种可以嵌入Web页面中的基于对象和事件驱动的解释性语言。

(2) JavaScript 是基于对象，Java 是面向对象。Java 是- -种真正的面向对象的语言，即使是开发简单的程序，也必须设计对象。JavaScript 是一种脚本语言，它可以用来制作与网络无关的，只与用户交互作用的复杂软件。它是一-种基 于对象和事件驱动的编程语言，因而它本身提供了非常丰富的内部对象供设计人员使用。

(3)两种语言在其浏览器中所执行的方式不一样。 Java的源代码在传递到客户端执行之前，必须经过编译，因而客户蟎上必须具有相应平台上的仿真器或解释器，它可以通过编译器或解释器实现独立于某个特定的平台编译代码的束缚。JavaScript 是- -种解释性编程语言，其源代码在发往客户端执行之前不需经过编译，而是将丈本格式的字符代码发送给客户，由浏览器解释执行。

(4)两种语言所采取的变量是不一样的。Java 采用强类型变量检查。JavaScript中变量声明，采用其弱类型。即变量在使用前不需进行声明，而是解释器在运行时检查其数据类型

(5)代码格式不样。Java 是一一种与HTML无关的格式，必须通过像HTML中引用外媒体那样进行装载，其代码以字节代码的形式保存在独立的文档中。JavaScript 的代码是一种文本字符格式，可以直接嵌入HTML文档中，并且可动态装载。编写HTML文档就像编辑文本文件样方便。

(6)嵌入方式不一样。在HTML文档中，两种编程语言的标识不同，JavaScript 使用<script>...</script>来标识，而Java使用<applet> .. </applet>来标识。

(7)静态绑定和动态绑定。Java 采用静态联编，即Java的对象引用必须在编译时进行，以使编译器能够实现强类型检查。JavaSeript 采用动态联编，即JavaScrpit的对象引用在运行时进行检查，如不经编译则无法实现对象引用的检查。

4 .JavaScript 在网页中的用法

JavaScript加入网页有两种方法。

(1)直接加入HTML文档。这是最常用的方法，大部分含有JavaSeript的网页都采用这种方法<script>...</script>. 而<script language-"Javascript" >用于告诉浏览器这是用JavaScript编写的程序，需要调动相应的解释程序进行解释。HTML的注释标签<1-->，用来去掉浏览器所不能识别的JavaScript 源代码，这对不支持JavaSeript语言的浏览器来说是很有用的。

//-Javascript结束:双斜杠表示JavaScript的注释部分，即从/开始到行尾的字符都被忽略。至于程序中所用到的document. write0函 数则表示将括号中的文字输出到窗口中去，这在后面将会详细介绍。另外，需要注意的是，<scp.../scrip> 的位置并不是固定的，可以包含在<head>...</head>或><body>...</body>中的任何地方。

(2)引用方式。如果已经存在一个JavaScript源文件(以js为扩展名)，则可以采用这种引用的方式，以提高程序代码的利用率。其中的url就是程序文件的地址。同样的，这样的语句可以放在HTMIL文档头部或主体的任何部分。如果要实现“直接插入方式”示例中的效果，可以首先创建一个JavaSeript 源代码文件“Scriptjs"。

**Servlet概述**

Servlet 是Sumn公司最早推出的基于Java基础来实现Web开发的一种技术，它已经存在很长一段时间了， Sun 公司将其囊括入最新的JavaEE API中。对于初学者而言，有必要了解一些与Servlet相关的知识，本章就从实用的角度来剖析Servlet的相关用法。

Servlet是以Java技术为基础的服务器端应用程序组件。已经出现很长时间了，先于J2EE平台出现。在过去的段时间内， Servlet曾经得到广泛的应用，目前，在J2EE项目开发中仍然广泛使用。现在Servlet已经是一种非常成熟的技术， 在后续章节中可以看到将JSP与Servlet 结合起来是很有效的。在Web应用程序架构中，它可以完全取代cGl程序。相比较而言，Serlet 具有下列优点。

执行效率高:每个Servlet 均对应一个独 立的“线程" (thread)，因此执行效率比cGI程序更高，也比较稳定。

进入门槛低: Servlet 是以Java为基础的，只要是熟悉Java的程序员就可以很轻易地编写出各式各样的Servlet组件。

拥有跨平台执行能力:借助Java本身的跨平台能力Servlet 得以运行在任何支持Java的平台上。关于Servlet相关的类，可以查看Sun公司提供的JavaEE API 文档，Sun 公司提供了有关Servlet的一系列标准。

Sun公司只定义了一系列的标准， 不同的厂商可以提供各自的标准实现，然后根据需要选择它的具体实现。在此我们使用的是tomet提供的Servlet 接口具体实现。这些实现类存放于tomeat目录下的lib文件夹下的srvet-apijar中。感兴趣的读者朋友可以将其解压比对一下API文档中的内容。

1.servlet生命周期

(1)加载Servlet类。可能在容器启动时发生，也可能在第一个客户使用时进行。

(2)实例化。Servlet 容器收到第一-个针 对此Servlet的请求时，调用该Servlet的默认构造方法，创建Servlet的一个实例。

(3)初始化。创建Servlet实例时，容器会调用该Servlet的init0方法(针对- -个Servlet 类只会调用一次)。

(4)服务。如果请求该Servlet,则容器启动一个新线程，并调用该Servlet的service0方法，由service0方法判断HTTP为什么请求，再调用相应的doGet(或doPotO方法。

(5)销毁。当Servlet容器判定某个Serlet实例不再需要时，在销毁实例之前会调用destroy0方法。(6)不可用。销毁实例并标记为垃圾收集。

2.生命周期中的3大重要时刻

(1)初始化

Servlet实例被创建之后，并在Servlet为客户请求提供服务之前，容器会为此Servlet创建一个ServletConfig对象，再调用init0方法。

ServletConfig对象也叫做Servlet起始参数对象，其含义如下。

当Servlet实例被创建之后，并在Servlet能为客户请求提供服务之前，容器会为此Servlet创建一-个ServletConfig对象。每个Servlet都有一个ServletConfig对象。用于向Servlet 传递部署时信息( Servlet 初始化参数)，而不必把这个信息硬编码到Servlet中。用于访问ServletContext.参数在部署描述文件中配置。

ServletContext对象也叫做Web应用起始参数对象，其含义如下。

当Servlet 容器初始化某个Web应用程序，它会为这个应用程序创建- -个ServletContext对象。每个Web应用只有一个ServletContext对象。用于访问Web应用的起始参数。用于得到服务器信息，包括容器的名字和版本，以及所支持API的版本等。

(2)服务

当第-一个客户请求到来时，容器会开始-一个新线程，调用srice0方法，而srvice0方法根据请求的HTTP方法(GET、 POST 等)来调用doGet0或doPost0.因此写Sevlet类- -般就是重写doGet(或doPost0方法来处理客户的请求。

(3)销毁

如果Servlet容器判断某个Servlet实体已经没有存在的必要了，或是Sevler容器本身必须被关闭时，Servlet 容器就会调用Servlet的dstor0方法。释放此Serlet实例所占用的存储器空间和相关资源。

**JSP概述**

JSP (Java Server Pages)是由Sun Microsytems公司倡导，许多公司参与一起建立的一种动态网页技术标准。JSP 技术是用Java语言作为脚本语言，JSP 网页为整个服务器端的Java库单元提供了一一个接口来服务于HTTP的应用程序。

JSP(Java Server Pages)其在动态网页的建设中有其强大而特别的功能。JSP与Microsoft的ASP技术非常相似。两者都提供在HTML代码中混合某种程序代码、由语言引擎解释执行程序代码的能力。下面我们简单的对它进行介绍。JSP页面最终会转换成servlet。因而，从根本上，JSP页面能够执行的任何任务都可以用servlet来完成。然而，这种底层的等同性并不意味着servlet和JSP页面对于所有的情况都等同适用。问题不在于技术的能力，而是二者在便利性、生产率和可维护性上的不同。毕竟，在特定平台上能够用Java编程语言完成的事情，同样可以用汇编语言来完成，但是选择哪种语言依旧十分重要。

在传统的网页HIML文件(.htm," htma)中加入Java程序片段(Scriptlet) 和JSP标记(ag),就构成了JSP网页(\*jsp)。Web 服务器在遇到访问JSP网页的请求时，首先执行其中的程序片段，然后将执行结果以HTML格式返回给客户。程序片段可以操作数据库、重新定向网页以及发送E-mail等，这就是建立动态网站所需要的功能。所有程序操作都在服务器端执行，网络上传送给客户端的仅是得到的结果。对客户端浏览器的要求最低，可以实现无Plugin.无ActiveX、无Java Applet,甚至无Fame。和单独使用servlet相比，JSP提供下述好处：

JSP中HTML的编写与维护更为简单。JSP中可以使用常规的HTML：没有额外的反斜杠，没有额外的双引号，也没有暗含的Java语法。 能够使用标准的网站开发工具。即使是那些对JSP一无所知的HTML工具，我们也可以使用，因为它们会忽略JSP标签（JSP tags）。

JSP的优点如下:

对于用户界面的更新，其实就是由Web Server进行的，所以给人的感觉更新很快。所有的应用都是基于服务器的，所以它们可以时刻保持最新版本。客户端的接口不是很繁琐，对于各种应用易于部署、维护和修改。

附录C 软件使用说明书

软件校企合作实训平台首页，系统管理员、企业教师、学生需输入用户名和密码才能进入系统，首页登录界面第三行是个下拉框，可以选择登录身份。只有学生、老师才需要注册，管理员信息由超级管理员管理。平台首页如图C-1所示：



图C-1 软件校企合作实训平台首页

系统管理员端包括用户管理、主题管理、教师教学任务管理、实训前材料管理、消息管理等模块；用户管理包含添加、删除用户、给教师学生修改用户信息。点击添加、修改、删除进行相关操作，详细查看学生详细信息。系统管理员端学生管理页面如图C-2所示：



图C-2 系统管理员端学生管理页面

管理员可管理实训平台所有人之间的消息记录。点击发送、修改、删除即可进行相关操作，详细即查看详情。系统管理员端消息管理页面如图C-3所示：



图C-3 系统管理员端消息管理页面

系统管理员端教师人数信息统计页面如图C-4所示：



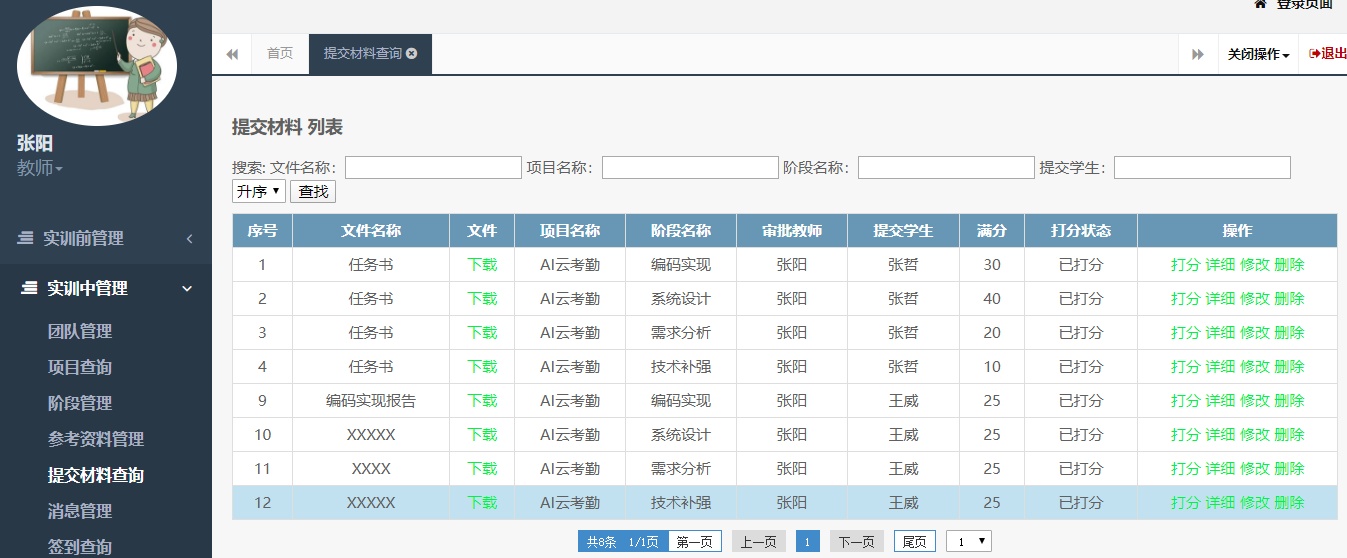
图C-4 系统管理员端教师人数信息统计页面

校企合作实训平台张阳教师端主页面如图C-5所示：



图C-5 校企合作实训平台张阳教师端主页面

教师端的实训中管理包括项目管理、团队管理、阶段管理、参考资料管理、提交材料管理、消息管理、查看考勤表。提交材料管理界面如图C-6所示：



图C-6 提交材料管理界面

教师端实训后管理有成绩管理，包括打分、查询、修改、删除成绩功能。教师可查询某位同学的具体项目具体阶段的成绩，也可进行相关修改删除操作。教师端成绩管理界面如图C-7所示：



图C-7 教师端成绩管理界面

学生端包括下载参考资料，提交上传材料，发送、浏览、删除消息，考勤管理（登录签到、请假、查看考勤表）功能，查看团队信息，查看项目信息，查看阶段信息，查看项目成绩等功能。校企合作实训平台张哲学生端主页面如图C-8所示：



图C-8 校企合作实训平台张哲学生端主页面

点击上传文件，选择文件，输入文件名称，即可完成上传提交材料操作。张哲同学学生端上传提交材料页面如图C-9所示：



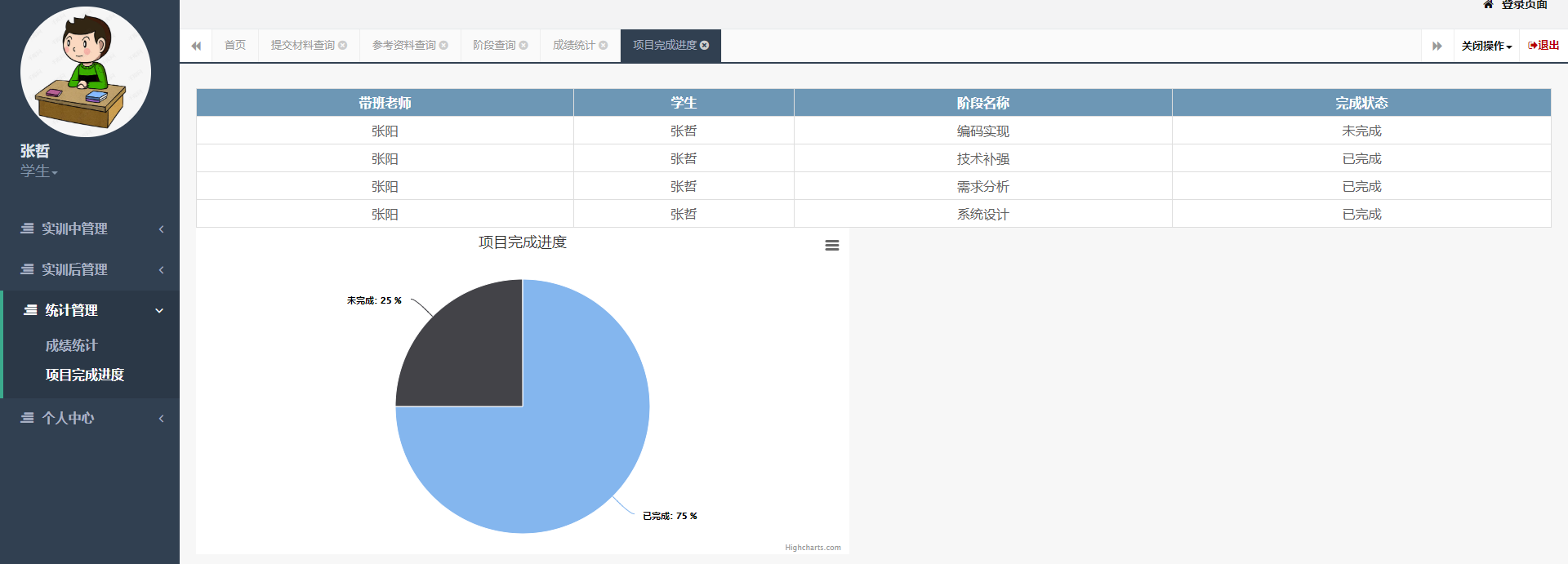
图C-9 张哲同学学生端上传提交材料页面

张哲同学学生端成绩统计页面如图C-10所示：



图C-10 张哲同学学生端成绩统计页面

张哲同学学生端项目完成进度统计页面如图C-11所示：



图C-11 张哲同学学生端项目完成进度统计页面

# 附录D 主要源代码

package com.spring.controller;

import com.spring.dao.TijiaocailiaoMapper;

import com.spring.entity.Tijiaocailiao;

import com.spring.service.TijiaocailiaoService;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.RequestMapping;

import tk.mybatis.mapper.entity.Example;

import util.Request;

import util.Info;

import dao.Query;

import java.util.\*;

import com.spring.entity.Jieduan;

import com.spring.service.JieduanService;

@Controller

public class TijiaocailiaoController extends BaseController

{@Autowired

private TijiaocailiaoMapper dao;

@Autowired

private TijiaocailiaoService service;

@Autowired

private JieduanService serviceRead;

/\*\*

\* 后台列表页

\*/

@RequestMapping("/tijiaocailiao\_list")

public String list()

{

if(!checkLogin()){return showError("尚未登录" , "./login.do");

}

String order = Request.get("order" , "id");

String sort = Request.get("sort" , "asc");

Example example = new Example(Tijiaocailiao.class);

Example.Criteria criteria = example.createCriteria();

String where = " 1=1 ";

where += getWhere();

criteria.andCondition(where);

if(sort.equals("desc")){

example.orderBy(order).desc();

}else{

example.orderBy(order).asc();

}

int page = request.getParameter("page") == null ? 1 : Integer.valueOf(request.getParameter("page"));

page = Math.max(1 , page);

List<Tijiaocailiao> list = service.selectPageExample(example , page , 12);

request.setAttribute("list" , list);

assign("orderBy" , order);

assign("sort" , sort);

assign("where" , where);

return "tijiaocailiao\_list";

}

public String getWhere()

{String where = " ";

if(Request.getInt("jieduanid")>0){

where += " AND jieduanid='"+Request.getInt("jieduanid")+"' ";

}

if(!Request.get("renwubianhao").equals("")) {

where += " AND renwubianhao LIKE '%"+Request.get("renwubianhao")+"%' ";

}

if(!Request.get("zhuti").equals("")) {

where += " AND zhuti ='"+Request.get("zhuti")+"' ";

}

if(!Request.get("xiangmumingcheng").equals("")) { where+="ANDxiangmumingchengLIKE'%"+Request.get("xiangmumingcheng")+"%' ";

}

if(!Request.get("tuanduimingcheng").equals("")) {

where += " AND tuanduimingcheng ='"+Request.get("tuanduimingcheng")+"' ";

}

if(!Request.get("jieduanmingcheng").equals("")) {

where+="ANDjieduanmingchengLIKE'%"+Request.get("jieduanmingcheng")+"%'";

}

if(!Request.get("manfen\_start").equals("")) {

where += " AND manfen >='"+Request.get("manfen\_start")+"' ";

}

if(!Request.get("manfen\_end").equals("")) {

where += " AND manfen <= '"+Request.get("manfen\_end")+"' ";

}

if(!Request.get("chuangjianren").equals("")) {

where += " AND chuangjianren LIKE '%"+Request.get("chuangjianren")+"%' ";

}

if(!Request.get("wenjian").equals("")) {

where += " AND wenjian LIKE '%"+Request.get("wenjian")+"%' ";

}

if(!Request.get("beizhu").equals("")) {

where += " AND beizhu LIKE '%"+Request.get("beizhu")+"%' ";

}

if(!Request.get("tijiaoren").equals("")) {

where += " AND tijiaoren LIKE '%"+Request.get("tijiaoren")+"%' ";

} return where;

}

@RequestMapping("/tijiaocailiao\_list\_chuangjianren")

public String listchuangjianren()

{

if(!checkLogin()){

return showError("尚未登录" , "./login.do");

}

String order = Request.get("order" , "id");

String sort = Request.get("sort" , "asc");

Example example = new Example(Tijiaocailiao.class);

Example.Criteria criteria = example.createCriteria();

String where = " chuangjianren='"+request.getSession().getAttribute("username")+"' ";

where += getWhere();

criteria.andCondition(where);

if(sort.equals("desc")){example.orderBy(order).desc(); }

else{example.orderBy(order).asc();}

int page = request.getParameter("page") == null ? 1 : Integer.valueOf(request.getParameter("page"));

page = Math.max(1 , page);

List<Tijiaocailiao> list = service.selectPageExample(example , page , 12);

request.setAttribute("list" , list);

assign("orderBy" , order);

assign("sort" , sort);

assign("where" , where);

return "tijiaocailiao\_list\_chuangjianren";}

@RequestMapping("/tijiaocailiao\_list\_tijiaoren")

public String listtijiaoren()

{ if(!checkLogin()){return showError("尚未登录" , "./login.do"); }

String order = Request.get("order" , "id");

String sort = Request.get("sort" , "asc");

Example example = new Example(Tijiaocailiao.class);

Example.Criteria criteria = example.createCriteria();

String where = " tijiaoren='"+request.getSession().getAttribute("username")+"' ";

where += getWhere();

criteria.andCondition(where);

if(sort.equals("desc")){

example.orderBy(order).desc();

}else{example.orderBy(order).asc(); }

int page = request.getParameter("page") == null ? 1 : Integer.valueOf(request.getParameter("page"));

page = Math.max(1 , page);

List<Tijiaocailiao> list = service.selectPageExample(example , page , 12);

request.setAttribute("list" , list);

assign("orderBy" , order);

assign("sort" , sort);

assign("where" , where);

return "tijiaocailiao\_list\_tijiaoren";

}

@RequestMapping("/tijiaocailiao\_add")

public String add()

{ int id = Request.getInt("id");

Jieduan readMap = serviceRead.find(id);

request.setAttribute("readMap" , readMap);

return "tijiaocailiao\_add"; }

@RequestMapping("/tijiaocailiao\_updt")

public String updt()

{int id = Request.getInt("id");

Tijiaocailiao mmm = service.find(id);

request.setAttribute("mmm" , mmm);

request.setAttribute("updtself" , 0);

return "tijiaocailiao\_updt";

}

/\*\*

\* 后台详情

\*/

@RequestMapping("/tijiaocailiao\_detail")

public String detail()

{int id = Request.getInt("id");

Tijiaocailiao map = service.find(id);

request.setAttribute("map" , map);

return "tijiaocailiao\_detail";

}

/\*\*

\* 删除

\*/

@RequestMapping("/tijiaocailiao\_delete")

public String delete()

{

if(!checkLogin()){return showError("尚未登录");}

int id = Request.getInt("id");

//delete\_before

service.delete(id);

return showSuccess("删除成功",request.getHeader("referer"));

}

}