

湖南工学院

2016 届毕业设计说明书

师生交流平台作业管理系统

院 、 部： 计算机与信息科学学院

学生姓名： 陈湘湘

指导教师： 王新祥 职称 副教授

专 业： 网络工程

班 级： 网络 1201

完成时间： 2016-5

摘 要

师生交流在教学过程中，甚至是整个校园都是不可或缺的一环。通过师生之间的交流，教师能够很明确的了解到当前的教学质量以及教学中存在的问题。近些年来，互联网的迅速发展极大地推动了教育事业的发展，许多面向教育教学领域的平台应运而生。他们以其和谐，轻松的氛围吸引了无数用户。针对这些特点，同时也是为紧跟时代发展的潮流，搭建一个在线师生交流平台是一件有意义，有价值的事情。本文将以此为背景，搭建基于 Web 的一个师生交流平台作业管理系统^[1]。

本系统采用 B/S 架构，结合 MVC 框架，结合多个前后端开发框架，并使用 MySQL 数据库，Tomcat 为服务器进行开发。具有可扩展，适应性强，易于维护等特点。

整个系统分为前台学生端，后台教师端和管理员端三部分。前台供学生使用，后台根据使用角色的不同分为教师端和管理员端两个部分。整个系统以社交化的风格，贴近学生生活，界面简洁友好，在相应的区域均有与之对应的提示信息。

本文首先将简单的介绍就本课题相关的背景，意义以及内容进行说明，接着对系统进行需求分析，梳理各个角色的关系，得到所需的角色用例图,然后着重阐述数据库设计、功能模块划分与详细。最后将简单的介绍系统的测试结果。

关键词： 作业管理；B/S 架构；Web；MySQL 数据库；MVC 框架

ABSTRACT

Communication between students and teachers is a necessary and desirable part in the whole campus, and even the whole teaching process. Through the communication, teachers can be very clear understanding of the current teaching quality and the existence of the problem. If only rely on the teacher to take the initiative to find students to talk the info.

In recent years, the rapid development of the Internet has greatly promoted the development of education, many platforms in the field of education and teaching came into being, numerous student users are attracted by their relaxed and harmonious atmosphere. According to these characteristics, and at the same time, to keep up with the trend of the times, it is a meaningful and valuable thing to create an online communication system for exchanging the mind between teachers and students. This paper takes this as the background. Guided by teachers, I will build a system which is based on Web for teachers and students to manage homework.

This system adopts B/S architecture, combined with the MVC framework. Combining multiple front end development framework, and use the MySQL database, Tomcat as the server for development. Is extensible, strong adaptability, easy maintenance, etc.

The system is divided into three parts: student terminal, teacher terminal and administrators' terminal. Front for student. According to different character of users, the background is divided into two parts, one of them is teacher terminal, The other one is administrators' terminal. The system in a social style, which is close to the student life, it contacts surface succinct friendly, function practical convenient.

This article will first introduce simply for the purpose of this topic related background, significance and content, and then I will analyze the demands of the system, combing the relationship between the different parts, getting the required role and using case diagram, then I focuses on stating function module division and database design in detail. Finally introduced simple system test results.

Key words : Homework management; B/S structure; JSP; MYSQL database;
MVC framework

目 录

1 前言.....	1
1.1 系统开发的背景.....	1
1.2 开发作业管理系统的意义与目的.....	1
1.3 作业管理系统的指导思想和内容.....	1
1.4 结构安排.....	1
2 开发环境、开发工具以及相关技术的概述.....	3
2.1 开发环境.....	3
2.2 开发工具简介.....	3
2.3 开发框架简介.....	3
2.4 界面编辑语言简介.....	4
2.5 数据库简介.....	4
2.6 服务器简介.....	5
3 系统需求分析.....	6
3.1 功能分析.....	6
3.1.1 学生端.....	6
3.1.2 教师端.....	7
3.1.3 管理员端.....	7
3.2 用例分析.....	8
3.3 非功能性需求.....	10
4 系统的设计.....	12
4.1 功能模块设计.....	12
4.1.1 学生端模块.....	12
4.1.2 教师端模块.....	12
4.1.3 管理员端模块.....	12
4.2 数据库设计.....	12
4.2.1 概念设计.....	13
4.2.2 逻辑设计.....	17
5 系统的实现.....	21
5.1 框架搭建.....	21

5.2 前端实现.....	21
5.2.1 学生端.....	21
5.2.2 教师端与管理员端.....	22
5.3 后端实现.....	25
5.3.1 学生端.....	25
5.3.2 教师端与管理员端.....	28
5.4 数据库配置.....	30
6 系统测试.....	32
6.1 创建和测试应用程序.....	32
6.2 测试内容.....	32
6.2.1 管理员端测试.....	32
6.2.2 学生端测试.....	34
参考文献.....	35
致 谢.....	36

1 前言

1.1 系统开发的背景

众所周知，随着计算机与互联网技术的飞速发展，人类已经步入了一个信息全球化的互联网时代。互联网时代的到来不光为人们提供了无数的便利，也正影响着各行各业的发展方向，2015年3月5日，李克强总理提出的“互联网+”行动计划，正是互联网时代在我国的一次集中体现。而网络信息技术的发展，离不开科学教育行业的支持，因此在被影响的各个行业中以教育行业最为显著。随着信息时代的来临，几乎各个学校都搭建起属于自己的平台用以教学和交流。但是这些平台都只是局限于学生被动的接收信息，学生的主动性不高，反响也不是特别好。作为教育的两大主体，教师与学生的交流方式还依旧停留在传统的交流方式上。因此搭建一个以相互交流为目的平台显得尤为重要^[15]。

1.2 开发作业管理系统的意义与目的

通过分析当前的师生交流方式不难看出，目前师生之间的交流仍旧停留在传统的方式上。但随着时代的发展，传统方式的不足也越来越明显，举例来说：首当其冲的是人力，物力，时间的资源浪费严重；其二，部分学生的情况不能体现整体情况，传统的师生交流方式收益不大；其三，间接的消息传递，不能很好的，很完整的完成信息传递。建立一个良好的师生交流平台，不仅能在一定程度上提高教学质量，而且还能够融洽师生关系，所以这一研究不仅仅是信息化教学环境的中一个重要环节，而且还应当将这以课题研究作为学校的重要基础设施来建设，这也有利于素质教育的实施。

1.3 作业管理系统的指导思想和内容

师生交流平台作业管理系统是一个以师生作业交流为核心，具有社交化风格的系统，本书介绍的是一个基于B/S架构（Browser/Server，浏览器/服务器）的，师生交流平台下的作业管理系统的研究与实现过程。说明书将从开发环境和开发工具谈起，对使用到的各类开发技术，框架，组件以及方法做简要的介绍，还将介绍MySQL数据库的实现、JSP实现动态网页以及为嫁接系统等做的一系列措施。最后将详细介绍如何开发一个基于Web的“作业管理系统”的全部过程^[2]。

1.4 结构安排

第一部分是前言部分，主要介绍本书的背景，目的、意义和系统开发的核心

思想与内容。

第二部分是开发环境、开发工具以及相关技术的概述，介绍了开发系统的环境要求，开发工具 MyEclipse 的使用，使用 HTML5+CSS3 和 JSP 混合的动态页面开发，Struts2、Spring、MyBatis 的在开发中的使用，使用 Tomcat 服务器进行系统的部署与测试，系统在开发过程中使用的 MySQL 数据库以及 DBCP 数据库连接池技术。

第三部分是针对作业管理系统的需求分析以及数据库模型设计，需求分析包括功能需求分析和非功能需求分析，根据需求设计与之对应的角色用例，并梳理角色间的关系。

第四部分是系统的概要设计，主要包括对系统的功能模块的划分，结合角色用例以及角色用例间的关系进行数据库的概念设计和逻辑设计。

第五部分是系统的详细设计和具体实现，包括系统的设计思路，界面设计，各功能模块的介绍以及部分核心代码展示等。

第六部分是系统的测试与展示，主要是根据系统需求逐步进行功能测试，并截图进行解说。

最后参考文献和致谢。

2 开发环境、开发工具以及相关技术的概述

2.1 开发环境

操作系统: Windows 10 x64 专业版

JDK 版本: 1.8.0

Maven 版本: 3.3.3

Tomcat 版本: 6.0.44

MySQL 版本: 5.5

2.2 开发工具简介

MyEclipse 是一款基于 Eclipse 的集成了众多企业级开发环境的开发工具, 由于其完整的支持众多开发技术以及各种主流开发框架, 并且允许用户自由定制和扩展的特性而被众多知名机构用于进行 Java、JavaEE 开发。本书介绍的作业管理系统, 正式一个 Web 应用系统, 涉及到的众多技术, MyEclipse 都能够完整的支持, 并且因为本系统隶属学生交流平台, 所以使用 MyEclipse 进行开发在合适不过了^[3]。

“Maven 是基于项目对象模型 (POM), 可以通过一小段描述信息来管理项目的构建, 报告和文档的软件项目管理工具”, 简单来说在就是一款项目管理工具, 利用其默认的项目构建方法, 并且使用 pom.xml 文件进行项目所需的依赖包管理, 省去人工寻找和下载程序所需的依赖包的问题。

2.3 开发框架简介

首先简单的介绍一下这三个框架:

Struts2 是一个基于 MVC 架构的 Web 应用程序框架, 其核心技术体现在拦截器方面, 是一个非常实用的, 由于其封装的众多视图技术以及少配置的理念, 能够满足快速开发的需求^[4]。

Spring 是一个用于解决企业应用开发的复杂性的框架, 适用于任何 Java 应用。其核心概念可以用: 面向切面 (AOP), 依赖注入 (DI), 控制反转 (IOC) 三个词来概括。在这里, 我们利用其容器的概念, 对整个项目中所涉及到的对象进行管理^[5]。

MyBatis 是一个非常年轻的, 同时也是非常优秀的持久层框架, 它代替了手动编写 JDBC 代码设置参数和结果集的麻烦, 使用简单的配置或注解方式, 就能将所要接口方法与普通 Java 对象映射成数据库中的记录。

使用这三个框架主要出于两方面的考虑：一、满足快速开发的要求；二、方便整个项目的管理与扩展。

2.4 界面编辑语言简介

将这四个放到一块讲，是因为他们都是前端技术。

HTML5: HTML5 是 HTML 的第五个版本，中文全称超文本标记语言 (Hyper Text Markup Language)，是标准通用标记语言下的一个应用。它是一个网页的内容组成部分，通过其规定的标记符号来显示所需的内容。

CSS3: CSS3 与 HTML5 一样，是 CSS 技术的一个升级版本，中文全称：层叠样式表 (Cascading Style Sheet)，它在一个网页中精准的控制着页面布局、字体、背景等样式，通常与 HTML 结合使用。

JavaScript: 是一种动态类型的、弱类型的、基于原型的语言。被广泛的应用于 Web 客户端开发，常用于为 HTML 网页增加动态功能^[7]。

JSP: JSP 是一种动态网页技术标准，中文全称：Java 服务器页面 (Java Server Pages)，它实现了 HTML 语法中的 Java 扩展，它与 Servlet 一样，是在服务器端指定的，而通常返回给客户端的是一个 HTML 页面，客户端可以通过浏览器浏览这种页面。这种过程实质上是 Java 代码的编译与运行^[8]。

在本系统中，并不是单一的使用某一种前端开发语言进行网页的编写，而是依据情况考虑，在适当的地方，采用适当的编程语言。出于执行效率的考虑，尽可能遵循“以 HTML 页面为主”的原则，仅在需要使用动态网页时才采用 JSP 编写。

2.5 数据库简介

MySQL 是一个开源的小型数据库管理系统，它以其体积小，速度快，成本低的特点，广泛的应用于中小型网站的数据库建设。在本系统中，出于对用户人数的考虑，最后决定使用 MySQL 数据库最合适^[6]。

DBCP: 中文全称为：数据库连接池 (Database Connection Pool)，如果使用最原始的 JDBC 操作的话，每次访问数据库都需要建立新的连接，而建立一个数据库连接又是非常耗时耗费资源的行为，而且考虑到潜在的高并发的可能，所以这种方法并不可取。因此采用了数据库连接池技术，它预先与数据库建立一定的连接，放入内存中，当应用程序需要访问数据库时，直接到连接池中申请一个就行，用完后再放回去，而且其参数可配置，连接数也是可变的，非常适合用于本系统的开发。

2.6 服务器简介

Tomcat 是一个轻量级的应用服务器，它运行时占用系统资源少，支持负载均衡和邮件服务等开发应用系统常用的功能，在中小型系统或者并发访问不是特别大的场合下被广泛使用，是开发和调试 JSP 程序的首选。

3 系统需求分析

3.1 功能分析

信息时代的来临，影响着传统的教学方式，由于获得的信息越来越多，传统的管理方式已经不堪重负，亟需一个数字化的平台来管理这些信息。作业作为教学过程中最多，同时也是最大的信息载体，承担着师生之间信息传递的重要媒介，它的管理，在师生交流平台必不可少的一部分。就目前的校园而言，传统的作业管理方式虽然能够满足需求，依旧有不少不足，但是信息化的作业管理系统恰恰弥补了传统方式的这些不足。所以，无论是适应信息化时代的要求，或是节约教学资源，还是弥补传统作业管理方式的不足，设计一个作业管理系统都是很有必要的。这里，通过对作业交流以及相关问题的研究和分析，我将作业管理系统分为三个用户群，前台为学生端，供学生使用，后台为教师端和管理员端，供教师和管理员使用，接下来，我将详细介绍这三大部分的功能^[9]。

3.1.1 学生端

学生端主要功能有：学生用户管理，作业交流，师生互动三大块功能。而他们分别可以细分为若干小功能，如：

用户登录验证：确保能根据学生的不同得到与之对应的信息。如，学生的个人作业信息，学生作业的评论内容等。

修改个人资料：在许可的范围内，为学生自主更新信息提供的一个途径。目前只允许修改个人联系方式和一次性的邮箱绑定操作。

密码维护：在本系统中，提供了三个密码维护方案：第一个方案，学生在能正常登陆的情况下自主修改密码；第二个方案，学生在忘记密码，但是已绑定邮箱的情况下，通过邮箱找回密码；第三个方案，学生忘记密码，且未绑定邮箱的情况下，由教师或管理员进行密码重置。

更新作业信息：系统提供两种更新个人作业信息的方式：第一种方式，登陆系统时，默认更新当前学生作业信息；第二种方式，成功登陆后，手动更新作业信息。两套方案确保学生作业信息时刻与所在班级的课程信息同步。

上交作业：学生根据课程的不同，选择默认格式的文件上交，当作业上交成功时，更新对应的作业状态，当作业上传失败，则不更新作业状态。

下载批改后的作业：学生根据作业状态判断作业是否已经被批改玩，当作业被批改且已由教师上传时，学生可下载批改过的作业进行查看。

评论作业：为了提高学生对作业的关注度，学生可以针对不同的作业进行评

论，作为教学质量的一个重要参考，评论信息可以被任何人可见。

匿名私信：专门为学生提供一个无压力的交流窗口，因为整个过程全匿名的状态下进行，所以学生不用有太多的顾虑，可以向老师说出想说的话。

3.1.2 教师端

教师端主要功能有：教师用户管理，学生信息维护，作业交流三大块。细分如下：

用户登录验证：根据验证结果区分管理员与教师权限，并进入不同的管理板块。

修改教师个人信息：在许可的范围内，为教师自主更新信息提供的一个途径。目前只允许修改个人联系方式和一次性的邮箱绑定操作。

密码维护：与学生找回密码一样，本系统中为教师维护密码提供了三个密码维护方案：第一种方案，教师在能正常登陆的情况下自主修改密码；第二种方案，教师在忘记密码，但是已绑定邮箱的情况下，通过邮箱找回密码；第三种方案，教师忘记密码，且未绑定邮箱的情况下，管理员进行密码重置。

学生信息维护：对应学生密码维护的第二、三种方案，并进行了加强。当学生在已绑定邮箱的情况下，教师可以根据学生请求进行绑定邮箱修改。同时教师也可以重置学生密码，协助学生找回密码。

布置作业：教师给自己执教班级的相应课程布置作业，是学生作业的来源。这一功能规定教师只能给自己教授课程下的班级布置作业，不能僭越。

下载批改作业：教师根据学生作业状态，选择已提交的作业下载，进行批阅。同时更新被批阅作业的学生的作业的状态，使学生能够及时得到信息反馈。

上传批改后的作业：当教师批改完作业后，将批改好的作业上传服务器，并且更新作业状态，使学生能够得到批改后的作业进行更正。

收集作业评论：教师发布的每一条作业，学生都可以进行评论，为了得到评论信息，教师可以选择对应的作业评论信息进行查看，使教师能够了解学生们对所教授的课程的掌握程度，并及时做出调整。

3.1.3 管理员端

管理员端主要有：学生管理，教师管理，班级管理，课程管理以及教授关系管理五大块功能。他们可以细分为：

学生管理：学生信息录入，新增，修改，删除，拥有所有操作的最高权限；

教师管理：教师信息录入，新增，修改，删除，拥有所有操作的最高权限；

班级管理：班级信息录入，新增，修改，删除，拥有所有操作的最高权限；

课程管理：课程信息录入，新增，修改，删除，拥有所有操作的最高权限；

班级-课程-教师关系管理：关联关系的录入，新增，修改，删除，拥有所有

操作的最高权限，这一功能中规定：一名教师在一个班级只允许教授一门课程；同理，一个班级的某一课程只允许由一名教师教授，三者关系相互依存，相互制约。

同时，在本系统中所有数据的删除操作均为逻辑删除，管理员删除操作也是一样。

3.2 用例分析

在上一小节，详细分析了作业管理系统三大部分的功能需求。学生端与教师端的作业交流为本系统的核心功能，其他的功能为支路功能，比如密码维护、匿名私信、作业评论等等。每个用户的登陆都会根据用户的角色不同而分配不同的权限，权限的不同，系统能提供的功能也不同。

学生端用例图如下：

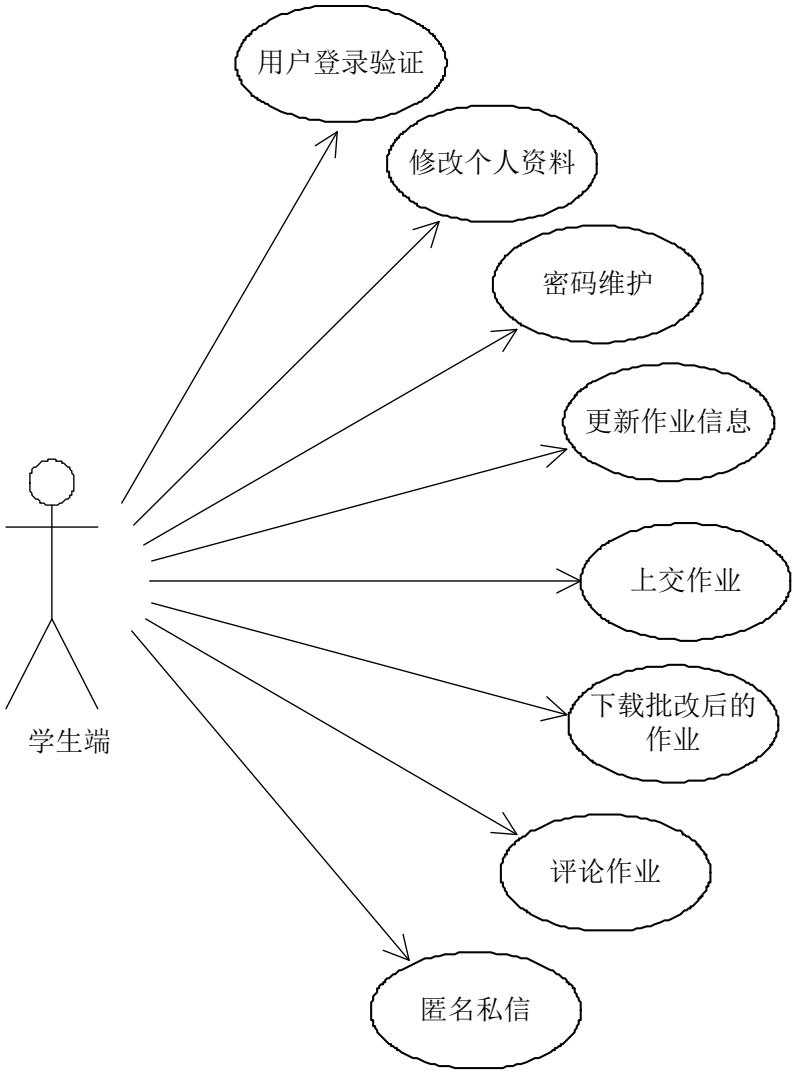


图 1 学生端用例图

教师端用例图如下：

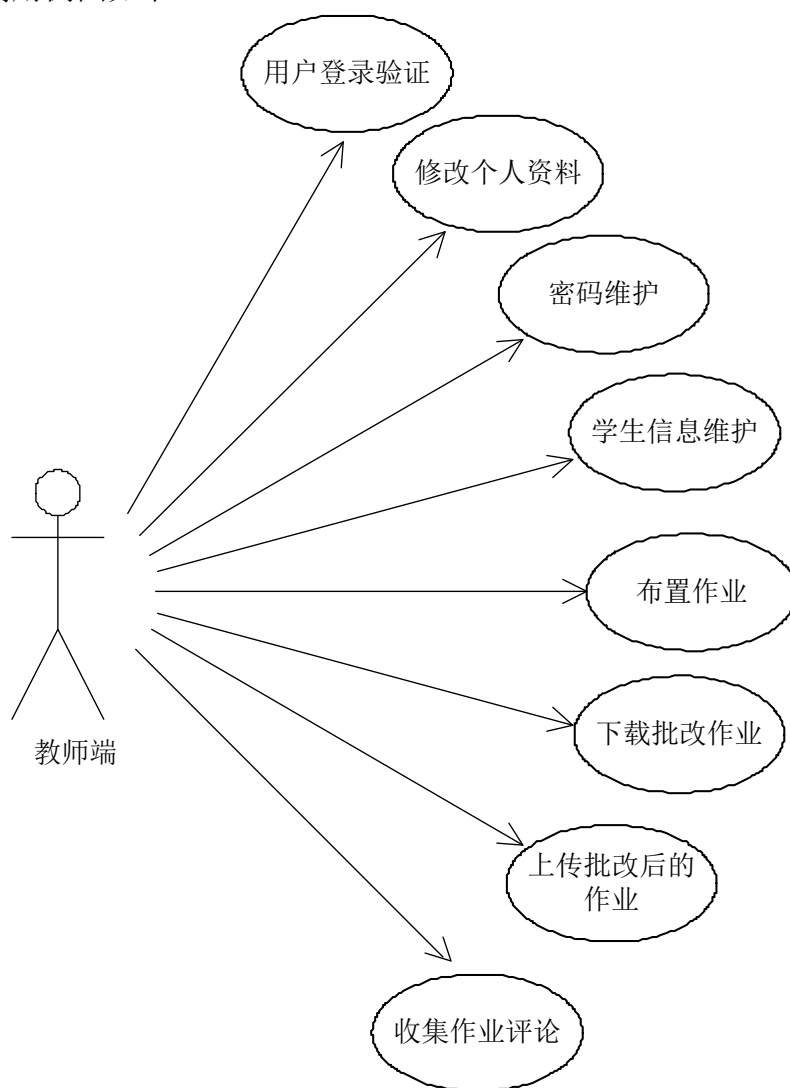


图 2 教师端用例图

管理员端用例图如下：

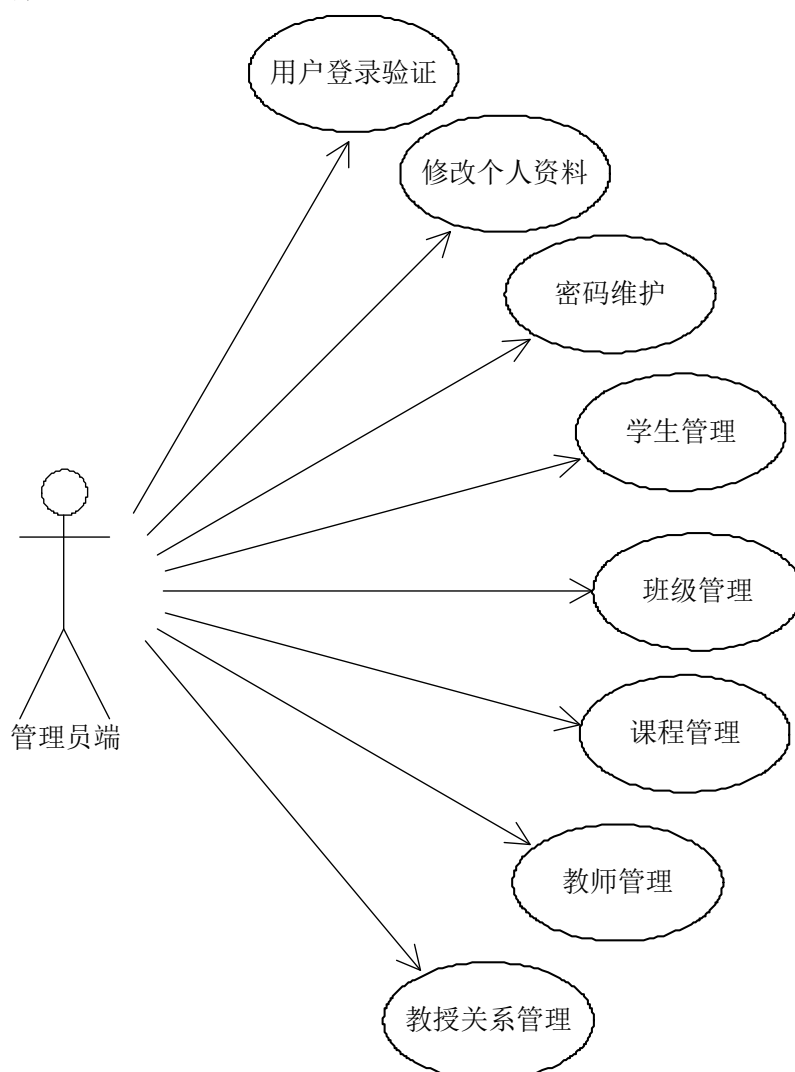


图 3 管理员端用例图

3.3 非功能性需求

本系统在设计数据库时，根据现实要求，添加了很多约束条件，因为这一设定，确保了数据在一定程度上的有效性，但是依然不存在不足。所以系统数据的录入须由专业人员或经过专业培训的人员进行数据录入。

在用户在使用过程中，与系统的任何交互操作，系统都应该能够及时的做出相应的信息反馈。而当发生系统内部错误时，也能针对错误给予用户及时的故障信息反馈。这一系列的信息反馈要求在 3s 内。

为了使系统的适应性更强，所以，在开发系统时，需要使用可以跨平台的计算机语言进行系统的开发或使用其他可以使系统能够跨平台使用的手段。

同时当遭遇突发故障时，要求系统能够安全保存用户信息，且在一定程度上保持系统功能的部分可用。同时在故障解除时，要求系统能够恢复被安全保存的

数据。

因为目前系统功能相对单一，所以为了对系统功能进行扩展，还要求系统拥有较强的灵活性、可扩展性。

除此之外，还要求系统具有操作简单的特性。无论从系统界面还是系统功能，都应以用户体验为出发点，方便用户能够在简单的阅读操作指南后便能独立的使用系统。

4 系统的设计

4.1 功能模块设计

4.1.1 学生端模块

综合上述各项需求分析，最后决定将学生端分为三大模块，他们分别是：学生信息管理模块、学生作业交流模块、师生互动模块。这三大功能模块又细分为八个功能点分别为：用户登录验证、修改个人资料、密码维护、更新个人作业信息、上交作业、下载批改后的作业、评论作业、匿名私信。学生用户拥有所提供的功能范围内的所有权限。其中密码维护功能、上交作业功能、下载批改后的作业功能、评论作业功能以及匿名私信功能，与教师端共享信息，且信息实时交互，由用户手动刷新。

4.1.2 教师端模块

综合上述各项需求分析，最后决定将教师端分为三大模块，他们分别是：教师用户管理模块，学生信息维护模块，作业交流模块。这三大功能模块又细分为八个功能点分别为：用户登录验证、修改个人资料、密码维护、学生信息维护、布置作业、下载并批改作业、上传批改后的作业、收集作业评论。教师用户拥有权限范围内的系统所提供功能的所有权限。其中作业功能模块为教师独有，拥有最高权限，不受管理员限制。收集作业评论功能，所收集的评论信息将以邮件的形式发送到指定邮箱中。

4.1.3 管理员端模块

综合上述各项需求分析，最后决定将管理员端分为两大模块，他们分别是：管理员用户管理模块，系统信息维护模块。这两大功能模块又细分为八个功能点分别为：用户登录验证、修改个人资料、密码维护、学生管理、班级管理、课程管理、教师管理、教授关系管理。管理员拥有系统内，除作业外的所有对象的最高操作权限。同时教授关系管理功能中所操作的数据必须为数据库已经存在的数据。

4.2 数据库设计

在本系统设计之初，充分以现实关系为基础，深入分析各角色与对象间的关系，考虑多种常见非常规情况等多方面因素。综合众多原因，考虑到数据关系繁琐，且数量会不断递增的情况，因此决定使用高效率的，易于维护的，且能与当

前系统对接的数据管理方式—数据库系统。同时，为了确保数据不被删除，对每一条数据均设计了状态列，用状态列的形式来标记数据的存在状态；为了避免数据冗余，针对每一条数据都设定了唯一的主键，在特殊的组合关系数据中，还使用到了外键关系；另外，为了加快数据速度，在可能被系统频繁调用的数据列上添加了索引；同时，对一些次要数据项，为了约束数据内容不为空，还设计了默认值，确保数据的有效。

4.2.1 概念设计

考虑到本系统管理的对象在现实中拥有众多属性以及复杂的关系，因此在设计之初，我决定使用数据模型的方式来构建数据对象，并针对需求进行抽象，抽取出对系统有用的属性，简化得到在本系统中有用的对象关系。目前在数据库设计过程中，使用较为广泛的数据模型有两种，分别是“概念数据模型”和“结构数据模型”。实体关系模型是一种从现实对象上抽取出试题类型与实体间的关系的数据模型构建方式，本系统在设计时，采用的就是“实体关系模型”（E-R 模型）。在构建好实体关系模型后，将可以使用 E-R 图的方式来表示数据模型。

E-R 图由三种基本图形组成：
矩形，代表实体对象（即被抽象的现实对象）；
菱形，代表实体对象间的关联方式（即现实中各个对象间的关系）；
椭圆形，代表实体对象具备的属性（即从现实对象所具备的属性中抽象出来的对系统有用的数据）。

各实体对象及其属性的实体属性图如下：
学生信息（学生 ID，班级 ID，学生姓名，学生密码，学生性别，学生年龄，联系方式，学生邮箱，学生状态，数据状态），如图 1 所示。

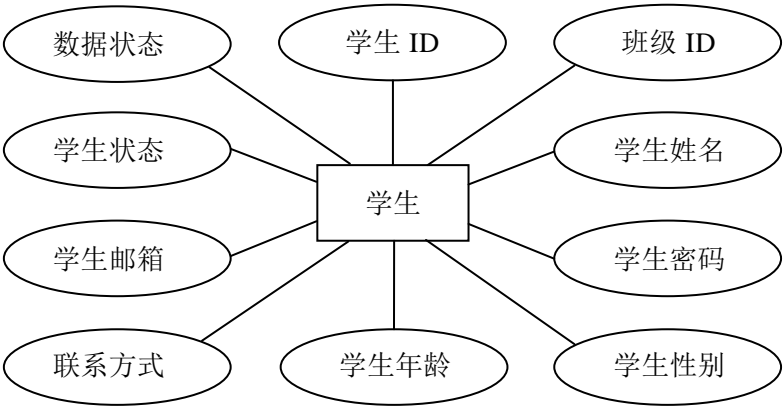


图 1 学生实体属性图

教师信息（教师 ID，教师姓名，教师密码，教师性别，教师年龄，联系方式，教师邮箱，教师状态，数据状态），如图 2 所示。

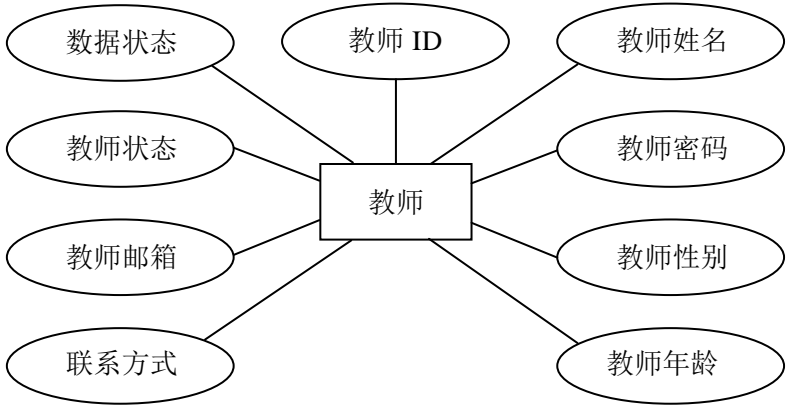


图 2 教师实体属性图

班级信息（班级 ID，班级名，班级人数，班级状态，数据状态），如图 3 所示。

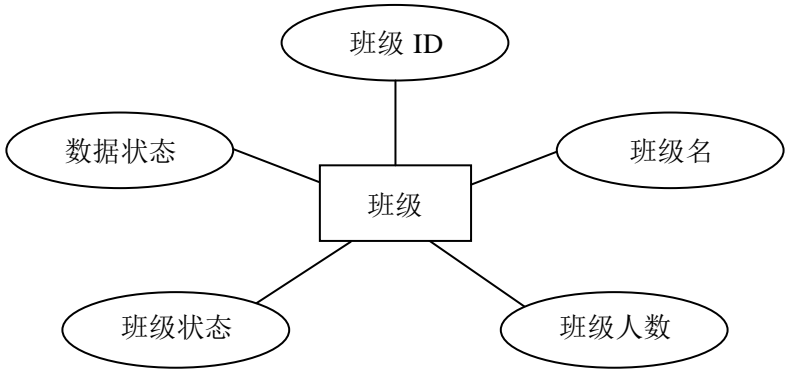


图 3 班级实体属性图

课程信息（课程 ID，课程名，课程状态，数据状态），如图 4 所示。

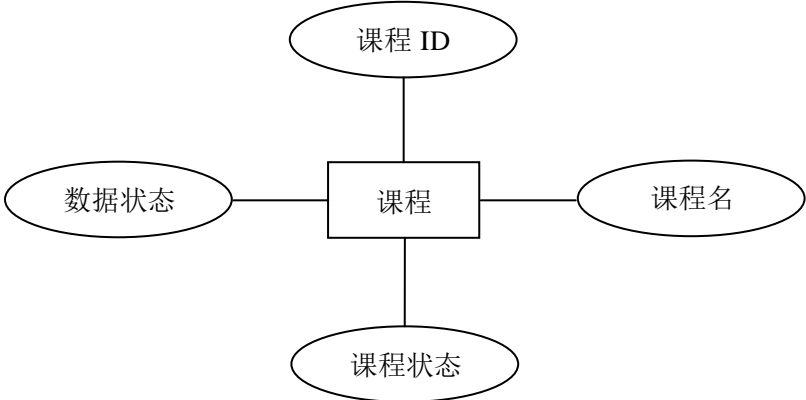


图 4 课程实体属性图

教授关系信息（教授关系 ID，班级 ID，课程 ID，教师 ID，教授关系状态，数据状态），如图 5 所示。

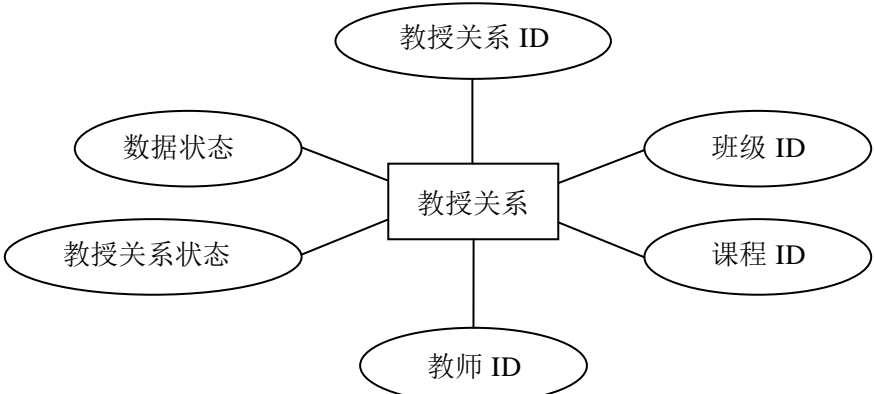


图 5 教授关系实体属性图

作业发布信息（作业发布 ID，班级 ID，课程 ID，作业名，创建时间，作业详情，作业状态，数据状态），如图 6 所示。

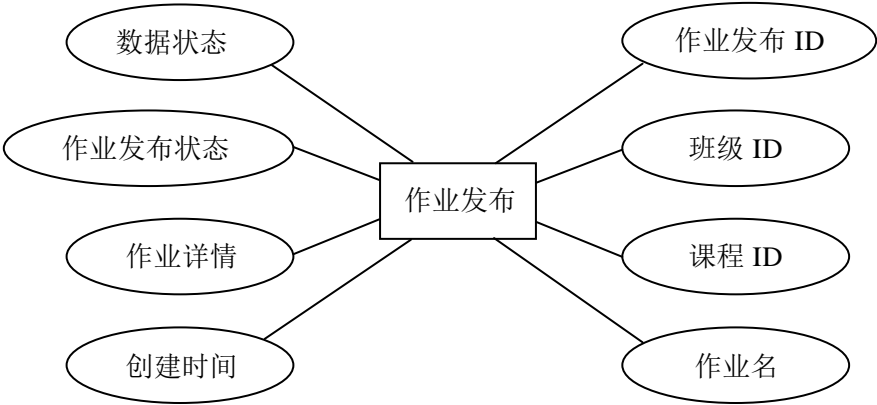


图 6 作业发布实体属性图

学生作业信息（学生作业 ID，学生 ID，作业发布 ID，作业上传路径，学生作业状态，数据状态），如图 7 所示。

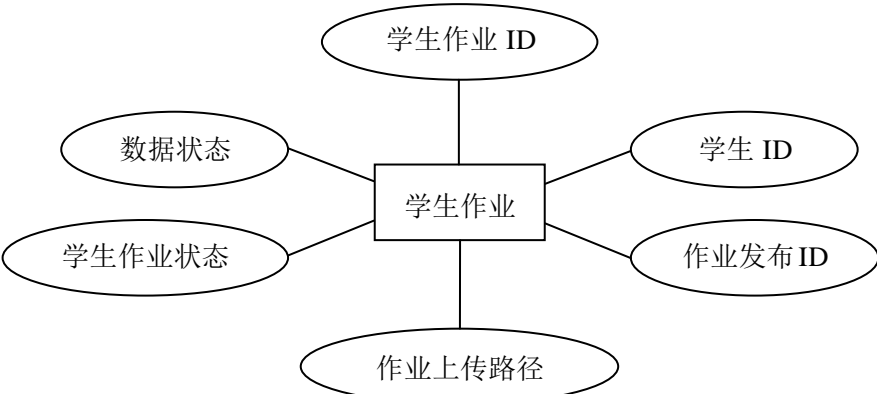


图 7 学生作业实体属性图

作业评论信息（评论 ID，被评论作业 ID，评论内容，评论时间，评论状态，数据状态），如图 8 所示。

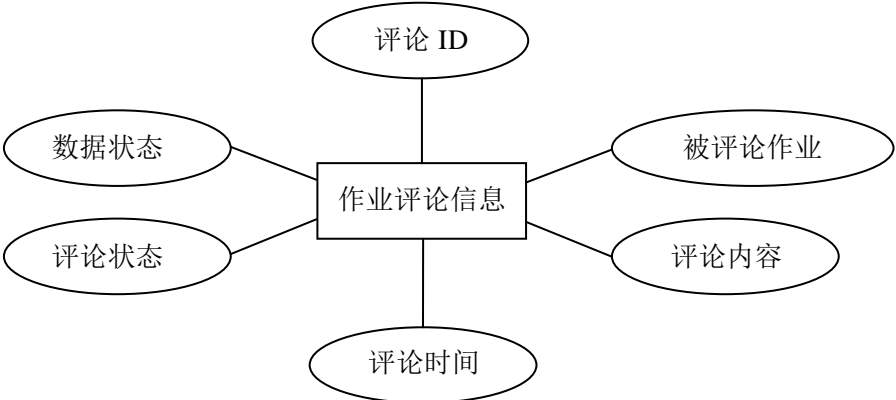


图 8 作业评论实体属性图

管理员信息（管理员 ID，管理员姓名，管理员密码，管理员性别，管理员年龄，管理员联系方式，管理员邮箱，管理员状态，数据状态），如图 9 所示。

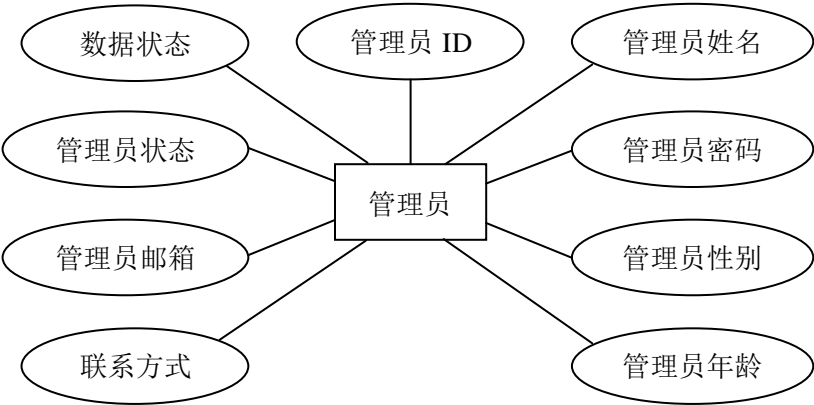


图 9 管理员实体属性图

4.2.2 逻辑设计

学生表 (students): 包含 10 个字段, 主键 StuId, 为自增列, 外键 ClaId, 关联班级表的 ClaId, 学生姓名 StuName, 学生密码 StuPwd, 非空约束, 默认值为 “123456”, 学生性别 StuSex, 非空约束, 默认值为 “男”, 学生年龄 StuAge, 非空约束, 默认值为 “18”, 学生联系方式 StuTel, 学生邮箱 StuEmail, 学生状态 StuStatus, 非空约束, 默认值为 “在读”, 数据状态 status, 非空约束, 默认值为 “有效”。在这张表由一些区分学生的标志属性构成, 其中, 学生编号, 同时也是前台登录账号。如表 1 所示:

表 1 学生表

字段名	数据类型	长度	约束	默认值	注释
StuId	int	20	primary key		学生 ID
ClaId	int	20	foreign key		班级 ID
StuName	varchar	20	not null		姓名
StuPwd	varchar	20	not null	123456	密码
StuSex	varchar	10	not null	男	性别
StuAge	int	3	not null	18	年龄
StuTel	int	11			联系方式
StuEmail	varchar	20			邮箱
StuStatus	varchar	10	not null	在读	学生状态
status	vachar	10	not null	有效	数据状态

教师表 (teacher): 包含 9 个字段, 主键 TechId, 为自增列, 教师姓名 TechName, 教师密码 TechPwd, 非空约束, 默认值为 “123456”, 教师性别 TechSex, 非空约束, 默认值为 “男”, 教师年龄 TechAge, 非空约束, 默认值为 “22”, 教师联系方式 TechTel, 教师邮箱 TechEmail, 教师状态 TechStatus, 非空约束, 默认值为 “在职”, 数据状态 status, 非空约束, 默认值为 “有效”。如表 2 所示:

表 2 教师表

字段名	数据类型	长度	约束	默认值	注释
TechId	int	20	primary key		教师 ID
TechName	varchar	20	not null		姓名
TechPwd	varchar	20	not null	123456	密码
TechSex	varchar	10	not null	男	性别
TechAge	int	3	not null	22	年龄
TechTel	int	11			联系方式

续表 2

TeachEmail	varchar	20			邮箱
TeachStatus	varchar	10	not null	在职	教师状态
status	vachar	10	not null	有效	数据状态

班级表 (classes): 包含 5 个字段, 主键 **ClaId**, 为自增列, 班级名 **ClaName**, 班级人数 **StuNum**, 非空约束, 默认值为 “0”, 班级状态 **ClaStatus**, 非空约束, 默认值为 “已开设”, 数据状态 **status**, 非空约束, 默认值为 “有效”。如表 3 所示。

表 3 班级表

字段名	数据类型	长度	约束	默认值	注释
ClaId	int	20	primary key		班级 ID
ClaName	varchar	20	not null		班级名
StuNum	int	3	not null	0	班级人数
ClaStatus	varchar	10	not null	已开设	班级状态
status	vachar	10	not null	有效	数据状态

课程表 (courses): 包含 4 个字段, 主键 **CouId**, 为自增列, 课程名 **CouName**, 课程状态 **CouStatus**, 非空约束, 默认值为 “已开设”, 数据状态 **status**, 非空约束, 默认值为 “有效”。如表 4 所示。

表 4 课程表

字段名	数据类型	长度	约束	默认值	注释
CouId	int	20	primary key		课程 ID
CouName	varchar	20	not null		课程名
CouStatus	varchar	10	not null	已开设	课程状态
status	vachar	10	not null	有效	数据状态

教授关系表 (Teach_Cla_Cou): 包含 6 个字段, 主键 **TccId**, 为自增列, 外键 **TeachId**, 与教师表的 **TeachId** 关联, 外键 **ClaId**, 与班级表的 **ClaId** 关联, 外键 **CouId**, 与课程表的 **CouId** 关联, 教授状态 **TccStatus**, 非空约束, 默认值为 “已分配”, 数据状态 **status**, 非空约束, 默认值为 “有效”。其中教师 ID、班级 ID、课程 ID 三个字段组合为一个唯一索引, 用来表示同一课程在同一班级只能由一名老师任教。如表 5 所示。

表 5 教授关系表

字段名	数据类型	长度	约束	默认值	注释
-----	------	----	----	-----	----

续表 5

TccId	int	20	primary key		教授关系 ID
TeachId	int	20	foreign key		教师 ID
ClaId	int	20	foreign key		班级 ID
CouId	int	20	foreign key		课程 ID
TccStatus	varchar	10	not null	已分配	教授状态
status	vachar	10	not null	有效	数据状态

作业发布表 (HworkPublish): 包含 7 个字段, 主键 HwpId, 为自增列, 外键 ClaId, 与班级表的 ClaId 关联, 外键 CouId, 与课程表的 CouId 关联, 作业名 HwpName, 作业信息 HwpInfo, 作业发布时间 HwpCreateTime, 作业发布状态 HwpStatus, 非空约束, 默认值为“已发布”, 数据状态 status, 非空约束, 默认值为“有效”。如表 6 所示。

表 6 作业发布表

字段名	数据类型	长度	约束	默认值	注释
HwpId	int	20	primary key		作业发布 ID
ClaId	int	20	foreign key		班级 ID
CouId	int	20	foreign key		课程 ID
HwpName	varchar	20	not null		作业名
HwpInfo	varchar	100			作业内容
HwpCreateTime	varchar	20	not null		发布时间
HwpStatus	varchar	10	not null	已发布	作业发布状态
status	vachar	10	not null	有效	数据状态

学生作业表 (Homework): 包含 5 个字段, 主键 HwId, 为自增列, 外键 StuId, 与学生表的 StuId 关联, 外键 HwpId, 与作业发布表的 HwpId 关联, 作业上传路径 HwPath, 学生作业状态 HwStatus, 非空约束, 默认值为“未完成”, 数据状态 status, 非空约束, 默认值为“有效”。如表 7 所示。

表 7 学生作业表

字段名	数据类型	长度	约束	默认值	注释
HwId	int	20	primary key		学生作业 ID
StuId	int	20	foreign key		学生 ID
HwpId	int	20	foreign key		作业发布 ID
HwPath	varchar	100			存储路径
HwStatus	varchar	10	not null	未完成	学生作业状态
status	vachar	10	not null	有效	数据状态

作业评论表 (HworkPublish): 包含 6 个字段, 主键 CommId, 为自增列, 外键 HwpId, 与作业发布表的 HwpId 关联, 评论内容 CommInfo, 评论时间 CommTime, 评论状态 CommStatus, 非空约束, 默认值为“未接收”, 数据状态 status, 非空约束, 默认值为“有效”。如表 8 所示:

表 8 作业评论表

字段名	数据类型	长度	约束	默认值	注释
CommId	int	20	primary key		评论 ID
HwpId	int	20	foreign key		作业发布 ID
CommInfo	varchar	100			评论内容
CommTime	varchar	20	not null		评论时间
CommStatus	varchar	10	not null	未接收	评论状态
status	vachar	10	not null	有效	数据状态

管理员表 (administrator): 包含 9 个字段, 主键 AdminId, 为自增列, 管理员姓名 AdminName, 管理员密码 AdminPwd, 非空约束, 默认值为“123456”, 管理员性别 AdminSex, 非空约束, 默认值为“男”, 管理员年龄 AdminAge, 非空约束, 默认值为“22”, 管理员联系方式 AdminTel, 管理员邮箱 AdminEmail, 管理员状态 AdminStatus, 非空约束, 默认值为“在职”, 数据状态 status, 非空约束, 默认值为“有效”。如表 9 所示:

表 9 管理员表

字段名	数据类型	长度	约束	默认值	注释
AdminId	int	20	primary key		管理员 ID
AdminName	varchar	20	not null		姓名
AdminPwd	varchar	20	not null	123456	密码
AdminSex	varchar	10	not null	男	性别
AdminAge	int	3	not null	18	年龄
AdminTel	int	11			联系方式
AdminEmail	varchar	20			邮箱
AdminStatus	varchar	10	not null	在职	管理员状态
status	vachar	10	not null	有效	数据状态

5 系统的实现

5.1 框架搭建

为了在兼顾的适应性、可扩展性以及工作性能的同时，又满足快速开发的要求，决定采用的整合了 Struts2+Spring+MyBatis 的 Maven-webapp 工程来开发本系统。此处略去环境搭建以及工程创建等步骤，直接从空白的 Maven-webapp 工程开始讲起。

为了能够使用三个框架带来的便利，首先，需要在工程的 pom.xml 文件中配置与之相关的 jar 包，在本系统中使用的各个框架版本分别为 Struts2-2.3.28，Spring-4.1.4.RELEASE，MyBatis-3.2.8，同时为了整合三个框架，还需导入 struts2-spring-plugin-2.3.28，mybatis-spring-1.2.1。因为使用的 MySQL 数据库，所以还需要导入 MySQL-connector-java-5.1.34，同时因为系统连接数据库使用的而是 DBCP 数据库连接池，所以还需导入 commons-dbcp-20030825.184428，commons-pool-20030825.183949，commons-collections-20040616。为了使前后台能以 JSON 数组的方式传参，需要导入 json-lib-2.4。最后，因为 Maven-webapp 工程中没有 JSP 的相依包，所以也需要手动导入，servlet-api-2.5，JSTL-1.2。在 web.xml 中做好 Struts2 前端拦截，同时在 struts.xml 中做好 Spring 托管。

至此一个空白的整合好 Struts2+Spring+MyBatis 的空白 Maven-webapp 工程就搭建好了，只需要写一个简单的测试用例即可测试框架是否搭建成功^[10]。

5.2 前端实现

5.2.1 学生端

为了使迎合学生的喜好，学生端的前端界面采用的是 bootstrap 风格。由于系统做了请求拦截，所以，未登录的用户默认访问的是登陆界面。如下图，采用的是 bootstrap 的模态框样式制作的一个登录窗口。

在界面编写的过程中，需要注意两个问题：第一个问题，jquery.js、bootstrap.css、bootstrap.js 都要引用，而且要特别注意 jquery.js、bootstrap.js 的顺序；第二个问题，模态框的样式中，一定要在最外层的容器的 class 样式中加入 show 样式。

在用户成功登陆后，将进入前台主界面，前台主界面主要分为四大块，个人信息展示，匿名私信，作业展示，工具栏。前台数据时根据用户的不同而变化的，所以在此处多次使用到了 JSTL 以及 Struts2 的标签库。在开发过程中，需特别注

意三个问题：第一个问题，在使用 JSTL 或者 Struts2 标签库时，一定要先在 JSP 页面开头申明引入对应标签库；第二个问题，除非经过特殊处理，否则 HTML 页面中是无法使用 JSTL 或者 Struts2 标签库的；第三个问题，JSTL 与 Struts2 标签库不允许嵌套使用。



图 10 前台登陆

5.2.2 教师端与管理员端

在管理后台，采用得是 easyui 风格，easyui 的使用与 bootstrap 是一样的，只需引入其定义好的 JS 与 CSS 文件即可使用其封装好的组件以及样式。easyui 拥有众多功能强大的组件，特别是其数据表格样式非常丰富，是作为管理后台界面设计时的首要选择。而且 easyui 的表格组件默认识别解析 JSON 格式的数据，所以有利于前后台之间以 JSON 格式传递数据。

管理后台在登陆成功后，将根据角色的不同而分配不同的管理权限，其主体界面如下：



图 11 前台主界面

学生交流平台作业管理系统后台

数据操作

教师区

管理员区

学生管理

班级管理

课程管理

教师管理

综合管理

数据查看

学生 ×

新增

修改

删除

所有

请输入查询条件

	<input type="checkbox"/>	学生编号	学生姓名	所属班级编号	所属班级名	性别	年龄	联系电话	邮箱	状态
1	<input type="checkbox"/>	1	李大妞	1	计科1201	女	19	15512315678	1234@haha.com	在读
2	<input type="checkbox"/>	2	赵狗蛋	1	计科1201	男	19	15512335678	1234@haha.com	在读
3	<input type="checkbox"/>	3	隔壁老王	1	计科1201	男	19	15515345678	1234@haha.com	在读
4	<input type="checkbox"/>	4	石榴姐	1	计科1201	男	22	15512375678	1234@haha.com	在读
5	<input type="checkbox"/>	5	罗玉凤	2	计科1202	女	22	15512345678	1234@haha.com	在读
6	<input type="checkbox"/>	6	芙蓉姐姐	2	计科1202	女	22	15812345678	1234@haha.com	在读
7	<input type="checkbox"/>	7	王蜜桃	2	计科1202	男	22	15517345678	1234@haha.com	在读
8	<input type="checkbox"/>	8	赵铁柱	2	计科1202	男	21	15514345678	1234@haha.com	在读
9	<input type="checkbox"/>	9	小岳岳	3	计科1203	男	21	15513345678	1234@haha.com	在读
10	<input type="checkbox"/>	10	王自健	3	计科1203	男	21	15512345878	1234@haha.com	在读

10

第 1 共 2 页

显示1到10,共18记录

图 12 后台主界面

5.3 后端实现

5.3.1 学生端

为了更清晰的了解系统的工作流程，我将以一次请求的顺序来介绍学生后端。

首先，是学生登录，因为每个学生用户所能够操作的数据是不同的，所以在登录时成功后，需要根据学生用户的不同进行查找。同时，系统需求分析中提出，用于在成功登陆之初会对名下作业进行更新，所以更新学生作业操作也需要在成功登陆后立即执行^[11]。

略去 struts2.xml 与 spring.xml 的相关配置^[12]，以及一系列的依赖注入的 get、set 方法，只介绍核心功能代码，代码示例如下：

登陆判断，位于 action 层。

```
public String front(){
    Students students = loginService.frontLogin(userId, userPwd);
    if (students != null) {
        homeworkService.updateHomework(students);
        List<StuHomeworkResultMap> list =homeworkService.
            selectAllHomeworkByStuId (students.getStuid());
        ActionContext.getContext().put("homework", list);
        ActionContext.getContext().put("user", students);
        return "front-success";
    } else {
        ActionContext.getContext().put("msg", "账号或密码错误，请核对");
        return "front-error";
    }
}
```

更新作业表操作，位于 service 层。

```
public void updateHomework(Students students) {
    // 对比班级作业数与学生作业数
    if(countHomeworkByStuId(students.getStuid()) !=
        countHworkPublishByClId(students.getClaid())) {
        // 不相等时，取出学生所有作业，班级所有作业进行比较
        List<Homework> stulist =selectStuHomeworkByStuId
            (students.getStuid());
```

```

List<HworkPublish> clalist = selectHworkPublishByClaId
    (students.getClaId());
// 循环取出班级作业记录中的作业编号，并于学生记录进行比较
Homework stu;
HworkPublish cla;
int j = 0;
for (int i = 0; i < clalist.size(); i++) {
    cla = clalist.get(i);
    if (stulist.size() > 0) {
        if (j < stulist.size()) {
            for (; j < stulist.size(); ) {
                stu = stulist.get(j);
                if (cla.getHwpid() == stu.getHwpid()) {
                    j++;
                    break;
                }
            }
        } else {
            Homework hw = new Homework();
            hw.setStuid(students.getStuid());
            hw.setHwpid(cla.getHwpid());
            homeworkMapper.insertSelective(hw);
        }
    } else {
        Homework hw = new Homework();
        hw.setStuid(students.getStuid());
        hw.setHwpid(cla.getHwpid());
        homeworkMapper.insertSelective(hw);
    }
}
}

```

文件上传，位于 action 层。

```
public String execute() {
    String rootpath = ServletActionContext.getServletContext()
        .getRealPath("/");
    String realpath = rootpath + savepath + uploadFileName;
    FileOutputStream fos = null;
    FileInputStream fis = null;
    try {
        fos = new FileOutputStream(realpath);
        fis = new FileInputStream(upload);
        byte[] buffer = new byte[1024];
        int length = 0;
        while ((length = fis.read(buffer)) != -1) {
            fos.write(buffer, 0, length);
        }
        Homework hw = new Homework();
        hw.setHwpid(Integer.parseInt(this.getHomeworkId()));
        hw.setStuid(Integer.parseInt(this.getStudentId()));
        hw.setHwpath(savepath + uploadFileName);
        hw.setHwstatus("已完成");
        homeworkService.updateHwPath(hw);
    } catch (FileNotFoundException e) {
        throw new RuntimeException("找不到文件", e);
    } catch (IOException e) {
        throw new RuntimeException("文件上传异常", e);
    } finally {
        try {
            fis.close();
        } catch (IOException e) {
            throw new RuntimeException("输入流关闭异常", e);
        }
        try {
            fos.close();
        } catch (IOException e) {
```



```

        throw new RuntimeException("输出流关闭异常", e);
    }
}
return "success";
}

```

文件下载，位于 action 层。

```

public String execute() {
    String rootpath = ServletActionContext.getServletContext().getRealPath("/");
    String realpath = rootpath + fileName;
    try {
        inputStream = new FileInputStream(realpath);
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
    return SUCCESS;
}

```

在这里，所有的 dao 操作均由 MyBatis 进行管理，MyBatis 的 Spring 托管将在之后进行介绍。

5.3.2 教师端与管理端

教师端与管理端的操作很相似，所以在此一起进行介绍。首先在选择需要操作的功能模块时，需要进行一个初始数据查询的操作，接下来就是对数据进行新增，修改，删除操作。所有操作均以 ajax 的方式与服务器进行数据交互^[13]。因为教师端与管理端的各项功能实现方式大同小异，所以此处选择不一一列举，仅以教授关系的新增操作为例进行介绍：

初始化数据操作，位于 action 层。

```

public String initData() {
    // 当前页，page 由分页工具负责传过
    int intPage = Integer.parseInt((page == null || page == "0") ? "1" : page);
    // 每页显示条数
    int pageSize = Integer.parseInt((rows == null || rows == "0") ? "10" : rows);
    // 每页的开始记录 第一页为 1 第二页为 number +1
    int StartNo = (intPage - 1) * pageSize;
    Map<String, Object> JSONMap = new HashMap<String, Object>();
    //定义 map
}

```

```

List<TccResultMap> list = tccInfoService.selectAllTccRM
    (StartNo,pageSize);
int total = tccInfoService.countAllTccRM();
//total 键存放总记录数，必须的，
//EasyUI 根据这个参数，可以计算 page 和 number 的值，
//这个值不是 users 的长度
JSONMap.put("total", total);
// rows 键 存放每页记录 list
JSONMap.put("rows", list);
// 格式化 result 一定要是 JSONObject
JSONObject = JSONObject.fromObject(JSONMap);
return SUCCESS;
}

```

当新增一条数据时，系统工作流程是：通过 action 层接收数据，调用相关 service 层对数据进行处理，而 service 层则根据需求调用 dao 操作的接口，service 层处理完成后，将数据返回给 action 层，再由 action 层封装数据返回给请求者。在本系统中 dao 接口由 MyBatis 提供实现，先关操作在此不详细介绍。

新增数据操作，位于 action 层。

```

public String addTccInfo() {
    int teachId = Integer.parseInt(this.getTeachId());
    int claId = Integer.parseInt(this.getClaid());
    int couId = Integer.parseInt(this.getCouId());
    teachClacou = new TeachClacou();
    teachClacou.setTeachid(teachId);
    teachClacou.setClaid(claId);
    teachClacou.setCoud(couId);
    int result = tccInfoService.insertTccInfo(teachClacou);
    Map<String, Object> JSONMap = new HashMap<String, Object>();
    JSONMap.put("addLine", result);
    JSONObject = JSONObject.fromObject(JSONMap);
    return SUCCESS;
}

```

新增数据 service 层实现，位于 service 层。

```
public int insertTccInfo(TeachClaCou teachClaCou) {  
    return teachClaCouMapper.insertSelective(teachClaCou);  
}
```

5.4 数据库配置

因为本系统采用的是基于配置文件的开发，所以在数据库配置中，也是以配置文件的方式来进行数据库以及数据库连接池的配置的^[14]。

首先，根据自己的需求建立数据库操作参数的配置文件。

数据库参数，位于 db.properties 文件中。

```
jdbc.url=jdbc:mysql://127.0.0.1:3306/ali?jdbcCompliantTruncation=false  
    &characterEncoding=UTF-8&allowMultiQueries=true  
jdbc.driver=com.mysql.jdbc.Driver  
jdbc.user=root  
jdbc.pwd=a
```

Spring 中的数据库以及数据库连接池的配置，位于 spring.xml 中。

```
<!-- 导入资源文件 -->  
<context:property-placeholder location="classpath: jdbc.properties" />  
<!-- 创建 dbcp 数据库连接池 -->  
<bean id="dataSource" class="org.apache.commons.dbcp.BasicDataSource"  
    destroy-method="close">  
    <property name="driverClassName" value="${jdbc.driver}" />  
    <property name="url" value="${jdbc.url}" />  
    <property name="username" value="${jdbc.user}" />  
    <property name="password" value="${jdbc.pwd}" />  
</bean>  
<!-- 加入 MyBatis 的支持，得到 Spring 创建的数据源，就能得到 session -->  
<bean id="sqlSessionFactory" class="org.mybatis.spring  
    .SqlSessionFactoryBean">  
    <property name="dataSource" ref="dataSource" />  
    <property name="mapperLocations" value="classpath*:com/ali  
        /mappers/*.xml" />  
</bean>  
<!-- 扫描映射 -->  
<bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">
```

```
<property name="basePackage" value="com.ali.mapper" />
<property name="sqlSessionFactoryBeanName" value=
    "sqlSessionFactory" />
</bean>
```

6 系统测试

6.1 创建和测试应用程序

因为使用的是 Maven 创建工程，所以，在后台可以单独的进行单元测试。
方法为：在工程下的“src/test/java”下，并根据需求创建所需的测试用例，用 Junit 进行测试即可，示例如下

```
public class TestTask {  
    public void HelloTest() throws Exception {  
        ApplicationContext applicationContext = new  
            ClassPathXmlApplicationContext  
                ("applicationContext-mybatis.xml",  
                "applicationContext-service.xml");  
        HomeworkService homeworkService = (HomeworkService)  
            applicationContext.getBean("homeworkService");  
        Students students = new Students();  
        students.setStuid(19);  
        students.setClaid(8);  
        homeworkService.updateHomework(students);  
    }  
}
```

6.2 测试内容

6.2.1 管理员端测试

(1) 教授关系管理初始数据查询。

新增 修改 删除 所有 请输入查询条件								
	<input type="checkbox"/>	记录编号	教师编号	教师姓名	班级编号	班级名	课程编号	课程名
1	<input type="checkbox"/>	4	1	赵日天	1	计科1201	4	C#语言程序设计
2	<input type="checkbox"/>	5	1	赵日天	2	计科1202	1	计算机导论
3	<input type="checkbox"/>	6	1	赵日天	2	计科1202	2	大学物理
4	<input type="checkbox"/>	7	1	赵日天	2	计科1202	3	C语言程序设计
5	<input type="checkbox"/>	8	1	赵日天	2	计科1202	4	C#语言程序设计
6	<input type="checkbox"/>	9	1	赵日天	3	计科1203	1	计算机导论
7	<input type="checkbox"/>	10	1	赵日天	3	计科1203	2	大学物理
8	<input type="checkbox"/>	11	1	赵日天	3	计科1203	3	C语言程序设计
9	<input type="checkbox"/>	12	1	赵日天	3	计科1203	4	C#语言程序设计
10	<input type="checkbox"/>	13	1	赵日天	4	信息1201	1	计算机导论

图 13 初始数据查询

(2) 新增教授关系。



新增教授关系对话框，包含三个输入框：教师编号、班级编号、课程编号，以及一个添加关系的按钮。

教师编号：
请输入教师编号

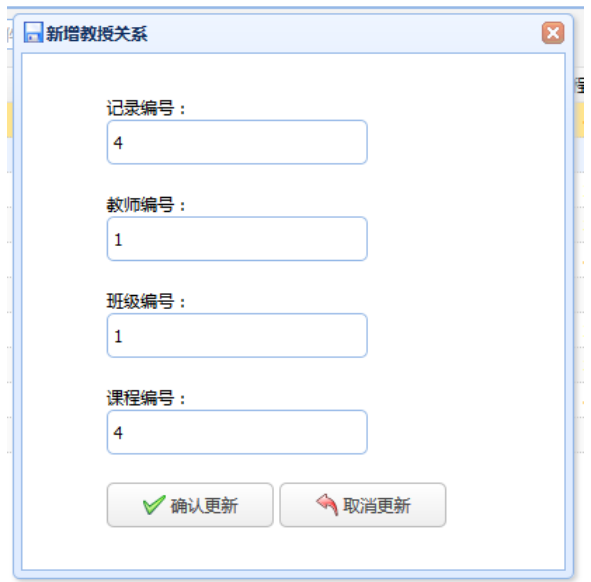
班级编号：
请输入班级编号

课程编号：
请输入课程编号

添加关系

图 14 新增教授关系

(3) 修改教授关系。



新增教授关系对话框，包含四个输入框：记录编号、教师编号、班级编号、课程编号，以及确认更新和取消更新的按钮。

记录编号：
4

教师编号：
1

班级编号：
1

课程编号：
4

确认更新 取消更新

图 15 修改教授关系

(4) 删除教授关系。

教师-班级-课程								
<div>新增 修改 删除 所有 请输入查询条件</div>								
	<input type="checkbox"/> 记录编号	教师编号	教师姓名	班级编号	班级名	课程编号	课程名	
1	<input checked="" type="checkbox"/>	5	1	赵日天	2	计科1202	1	计算机导论
2	<input type="checkbox"/>	6	1	赵日天	2	计科1202	2	大学物理
3	<input type="checkbox"/>	7	1	赵日天	2	计科1202	3	大学程序设计

图 16 删除教授关系

6.2.2 学生端测试

(1) 学生选查询课程下的作业。



图 17 学生选查询课程下的作业

(2) 提交作业。



图 18 提交作业

参考文献

- [1] 贺松平. 基于 MVC 模式的 B/S 架构的研究及应用[D]. 武汉: 华中科技大学, 2011
- [2] 赵俊峰, 等. Java Web 应用开发案例教程: 基于 MVC 模式的 JSP+Servlet+JDBC 和 AJAX[M]. 北京: 清华大学出版社, 2015. 14~307
- [3] 陈刚. Eclipse 从入门到精通[M]. 北京: 清华大学出版社, 2007. 88~275
- [4] 陈莉. 基于 Struts2 框架的应用研究[M]. 北京: 电子工业出版社, 2013. 65~163
- [5] 缪忠剑. 基于 Spring 的集成化 Web 开发平台的研究与实现[M]. 北京: 机械工业出版社, 2013. 12~225
- [6] 杜波依斯. MySQL 技术内幕[M]. (第 4 版). 北京: 人民邮电出版社, 2010. 255~321
- [7] Nicholas C Z. Java Script [M]. (第三版). 北京: 人民邮电出版社, 2015. 303~388
- [8] 赵朋飞. JSP 课程教学改革探讨[J]. 电脑知识与技术, 2014, 10(33): 7955-7956
- [9] 张海藩, 吕云翔. 软件工程[M]. (第四版). 北京: 人民邮电出版社, 2012. 125~208
- [10] Zoya Ali. Designing Object Oriented Software Applications within the Context of Software Frameworks[D]. Ohio State University, 2011
- [11] Metsker S J. Java 设计模式[M]. (第 2 版). 北京: 电子工业出版社, 2012. 303~320
- [12] Chuck Gavaness. Programming Jakarta Struts2nd Edition[J]. 北京: 电子工业出版社, 2011. 177~445
- [13] Ted Husted. Struts In Action[M]. 北京: 电子工业出版社, 2013. 116~241
- [14] Wu Xie, Huimin Zhang, Bizhong Wei, et al. Data mining of graduation project selection database Procedia Engineering[J]. 爱思唯尔期刊, 2011-2. 32~35
- [15] Rachit M G, YaminiSood, Balaji K, et al. A Framework Based Approach for the Development of Web Based Applications Waknaghat[J]. Jaypee University of Information Technology, 2011. 15~22

致 谢

时间飞逝，岁月如梭，一转眼我就要毕业了，回首过去的四年，心中倍感充实。而当我写完这份说明书的最后一个字时，一种如释重负的感觉油然而生，心中颇有感触。再次，首先感谢我的指导老师王新祥老师。感谢王老师在忙碌的教学工作中，抽出时间来为查看、批改我们的毕业设计。同时，也感谢大学四年里，我的所有的老师们，谢谢你们们的关心与教导。

最后，感谢四年来，和我一起成长的各位同学，感谢他们四年来给予我的支持与帮助，使我度过了充实的四年大学生活。