|  |  |
| --- | --- |
| **项目编号** |  |
| **文档编号** |  |
| **密级** |  |

**作业管理系统详细设计**

**V7.0**

**评审日期：2018年7月5日**

**编写人：黄灵专**

目录

[1．导言 4](#_Toc8344)

[1.1 目的 4](#_Toc13961)

[1.2 范围 4](#_Toc7160)

[1.3 引用标准 5](#_Toc14866)

[1.4 参考资料 5](#_Toc26626)

[1.5 版本更新信息 5](#_Toc11048)

[2．详细设计简述 6](#_Toc2498)

[2.1 设计简介 6](#_Toc2306)

[2.2 模块简介 6](#_Toc16518)

[3、界面详细设计 7](#_Toc9951)

[3.1、页面设计说明 7](#_Toc1810)

[3.2、静态页面详细设计 7](#_Toc16109)

[4.1、公用模块的详细设计 9](#_Toc6002)

[4.1.1 表现层 9](#_Toc12628)

[4.1.1 控制层 10](#_Toc20044)

[4.1.3 模型层 10](#_Toc30904)

[4.2、登录模块的详细设计 13](#_Toc29160)

[4.2.1 表现层 13](#_Toc26544)

[4.2.1 控制层 14](#_Toc16961)

[4.2.3 模型层 16](#_Toc7936)

[4.3、作业管理模块的详细设计 18](#_Toc8928)

[4.3.1 表现层 18](#_Toc27288)

[4.3.2 控制层 19](#_Toc29779)

[4.3.3 模型层 21](#_Toc25517)

[5、 管理员模块详细设计 23](#_Toc8816)

[5.1、添加用户模块详细设计 23](#_Toc26719)

[5.1.1 表现层 24](#_Toc21977)

[5.1.2 控制层 25](#_Toc15818)

[5.1.3 模型层 25](#_Toc27212)

[5.2、添加课程模块详细设计 26](#_Toc6010)

[5.2.1 表现层 26](#_Toc6341)

[5.2.2 控制层 27](#_Toc22420)

[5.2.3 模型层 28](#_Toc31625)

[5.3、删除用户模块详细设计 28](#_Toc13620)

[5.3.1 表现层 29](#_Toc900)

[5.3.2 控制层 30](#_Toc8912)

[5.3.3 模型层 30](#_Toc29094)

[6、公共部分模块详细设计 31](#_Toc23715)

[6.1 查看作业 31](#_Toc15397)

[6.2 登录页面 32](#_Toc1890)

[6.3 查看个人信息 34](#_Toc18124)

# 1．导言

## 1.1 目的

该文档是对作业管理系统概要设计进行的详细说明，是为了指导和规范作业管理系统开发而制定的详细开发设计文档。

本文档的预期读者包括：

* 设计开发人员
* 项目管理人员
* 测试人员
* 用户

## 1.2 范围

该文档内容涵盖作业管理系统的所有功能模块的详细设计描述。

## 1.3 引用标准

[1] 《软件工程案例教程 第2版》 韩万江等 机械工业出版社

[2] 《UML统一建模基础教程》 刘小松     机械工业出版社

[3] 《实用软件文档写作》   肖刚等     清华大学出版社

## 1.4 参考资料

[1] 《软件工程案例教程 第2版》 韩万江等 机械工业出版社

[2] 《UML统一建模基础教程》 刘小松     机械工业出版社

[3] 《实用软件文档写作》   肖刚等     清华大学出版社

## 1.5 版本更新信息

本文档的更新记录如下表1-1所示。

表1-1 文档更新记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 修改编号 | 修改日期 | 修改后版本 | 修改位置 | 修改内容概述 | 审核人 |
| 001 | 2018.6.28 | 1.0 | 全部 | 初始发布版本 | 韦东慧,阳长松 |
| 002 | 2018.6.29 | 2.0 | 部分 |  | 韦东慧,阳长松 |
| 003 | 2018.7.03 | 3.0 | 部分 |  | 韦东慧,阳长松 |
| 004 | 2018.07.03 | 4.0 | 部分 | 第六部分公共界面 | 韦东慧,阳长松 |
| 005 | 2018.07.03 | 5.0 | 部分 | 第五部分管理端模块详细设计 | 韦东慧,阳长松 |
| 006 | 2018.07.03 | 6.0 | 部分 | 第三部分界面详细设计 | 韦东慧,阳长松 |
| 007 | 2018.07.03 | 7.0 | 部分 | 第四部分客户端模块详细设计 | 韦东慧,阳长松 |

# 2．详细设计简述

## 2.1 设计简介

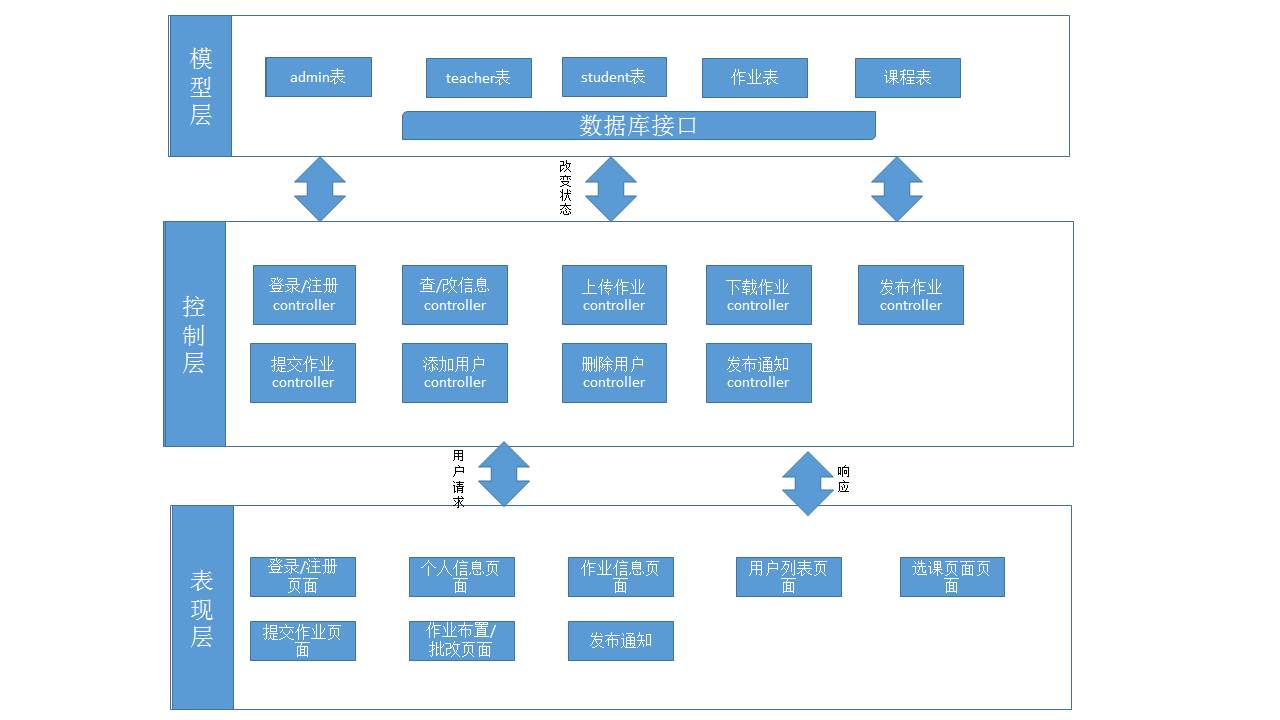
作业管理系统分层结构图如图2-1所示。

图 2-1 系统分层示意图

## 2.2 模块简介

系统功能模块图如图2-2所示。

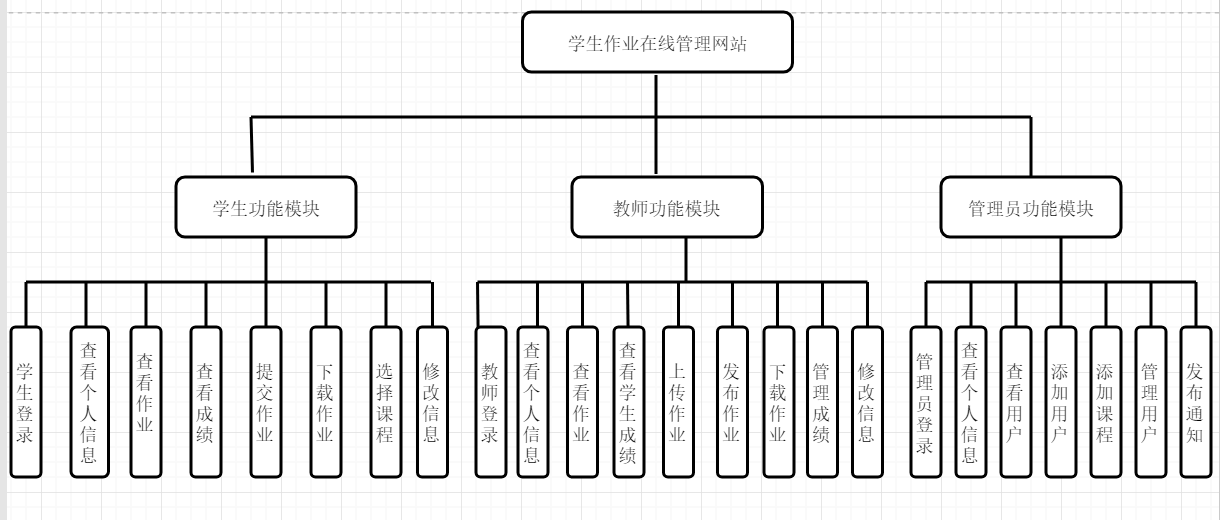


图2-2 系统功能模块图

# 3、界面详细设计

## 3.1、页面设计说明

本系统界面采用HTML+CSS+JSP技术来完成前端设计，坚持图形用户界面（GUI）设计原则，界面直观、对用户透明。用户接触软件后对界面上对应的功能一目了然、不需要多少培训就可以方便使用本应用系统。

## 3.2、静态页面详细设计

表3-1 页面表现层设计

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 登录界面 | Login.jsp | 用户进入系统的入口 |
| 主显示界面 | Homepage.jsp | 进入系统首先显示的界面 |
| 文件下载界面 | fileDowload.jsp | 文件下载 |
| 主索引界面 | index.jsp | 功能索引 |
| 学生用户索引界面 | index\_student.jsp | 显示学生功能 |
| 教师用户索引界面 | index\_teacher.jsp | 显示教师功能 |
| 用户名检查界面 | checkUserName.jsp | 检查用户名是否存在 |
| 教师检查界面 | check\_teacher.jsp | 检查教师用户名 |
| 布置作业界面 | fix\_homework.jsp | 布置作业各部分内容 |
| 获取教师信息界面 | getTeacher.jsp | 获取教师信息 |
| 作业批改界面 | homework\_fix.jsp | 作业批改 |
| 密码修改界面 | passworldchange.jsp | 修改密码 |
| 查看作业界面 | t\_homework.jsp | 查看作业 |
| 教师信息输入界面 | t\_info.jsp | 教师输入个人信息 |
| 教师登录界面 | teacherLogin.jsp | 教师登录 |
| 学生检查界面 | check\_student.jsp | 检查学生用户名 |
| 学生写作业界面 | do\_homework.jsp | 学生写作业 |
| 学生个人信息界面 | home.jsp | 修改密码 |
| 学生作业信息主页面 | home\_info.jsp | 显示学生作业信息 |
| 学生作业上传页面 | homework\_up.jsp | 作业上传 |
| 学生信息修改界面 | s\_info.jsp | 学生信息修改 |
| 学生作业提交页面 | uphomework.jsp | 作业提交 |
| 学生密码修改界面 | user\_fix.jsp | 学生密码修改界面 |
| 班级添加界面 | addClass.jsp | 班级添加 |
| 教师添加界面 | addTeacher.jsp | 教师添加 |
| 管理员添加界面 | adminLogin.jsp | 管理员添加 |
| 管理员账号检查界面 | check.jsp | 管理员账号检查 |
| 删除班级界面 | delClass.jsp | 删除界面 |
| 删除成功提醒界面 | delete.jsp | 删除成功提醒 |
| 删除教师成功提醒界面 | deletTeacher.jsp | 删除教师 |
| 查询老师信息界面 | findTeacher.jsp | 查询教师信息 |
| 删除老师信息界面 | teacherList.jsp | 显示删除教师信息界面 |

作业管理系统静态页面见表3-1。

4、客户端模块详细设计

客户端模块主要包括公用模块，登录模块，作业管理模块。

## 4.1、公用模块的详细设计

公用模块是指在系统在运行过程中将会被其他模块调用的基础资源。

### 4.1.1 表现层

公用模块的表现层主要是一些页面出错时的跳转页面，显示当前错误的结果处理报告。具体见表4-1。

表 4-1

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 登录出错页面 | loginError.jsp | 登入时出现严重错误时跳转到该页面。 |
| 统一异常页面 | exceptionPage.jsp | 在系统运行时出现未知错误时跳转到该页面。 |

### 4.1.1 控制层

**public** **class** login **extends** HttpServlet {}

### 4.1.3 模型层

基础数据库操作类DBbean.java代码结构

**public** **class** DBbean {

**public** **static** Connection getConn() {

Connection conn = **null**;

**try** {

**int** port=1433;

String db="jdbc:sqlserver://localhost:"+port+";DatabaseName=学生管理系统";

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

conn = DriverManager.*getConnection*(db,"sa","123456");

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** conn;

}

**public** **static** PreparedStatement getPrepareStatement(Connection conn,String sql)

{

PreparedStatement pstmt = **null**;

**try**

{

**if**(conn!=**null**){

pstmt = conn.prepareStatement(sql);

}

}**catch**(SQLException e)

{

e.printStackTrace();

}

**return** pstmt;

}

**public** **static** Statement getStatement(Connection conn) {

Statement stmt = **null**;

**try** {

**if**(conn != **null**) {

stmt = conn.createStatement();

}

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** stmt;

}

**public** **static** ResultSet getResultSet(Statement stmt, String sql) {

ResultSet rs = **null**;

**try** {

**if**(stmt != **null**) {

rs = stmt.executeQuery(sql);

}

} **catch** (SQLException e) {

e.printStackTrace();

}

**return** rs;

}

**public** **static** PreparedStatement prepareStmt(Connection conn,String sql)

{

PreparedStatement pstmt = **null**;

**try**

{

pstmt = conn.prepareStatement(sql);

}**catch**(SQLException e)

{

}

**return** pstmt;

}

**public** **static** **void** close(Connection conn) {

**try** {

**if**(conn != **null**) {

conn.close();

conn = **null**;

}

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **static** **void** close(Statement stmt) {

**try** {

**if**(stmt != **null**) {

stmt.close();

stmt = **null**;

}

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **static** **void** close(ResultSet rs) {

**try** {

**if**(rs != **null**) {

rs.close();

rs = **null**;

}

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **static** **void** close(PreparedStatement prepareStmt) {

**try** {

**if**(prepareStmt != **null**) {

prepareStmt.close();

prepareStmt = **null**;

}

} **catch** (SQLException e) {

e.printStackTrace();

}

}

// public static void main(String args[])

// {

// getConn();

// System.out.println("成功！");

// }

}：

## 4.2、登录模块的详细设计

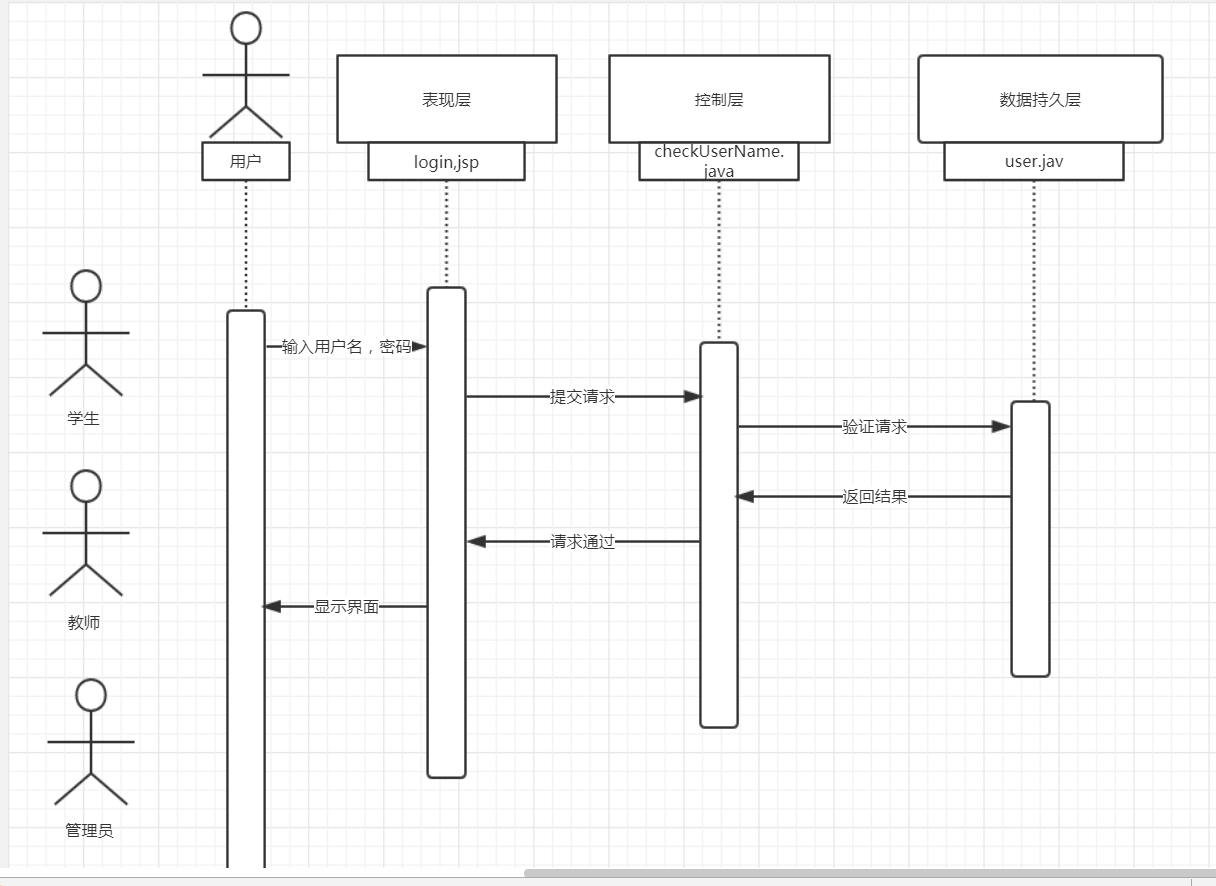


图 4-1 系统响应示意图

### 4.2.1 表现层

登录模块的表现层主要完成不同用户的登录功能，登录页面时要求用户输入账号密码的基本信息，确认后页面给出响应消息，提示登录成功或失败的提示。表现层对应的JSP页面列表见表4-2所示。

表 4-2 登入模块表现层JSP列表

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 登录页面 | login.jsp | 用户（教师、学生、游客）登入功能，当登入出错时给出提示。 |

login.jsp的流程图如图4-2所示。

进入游首页

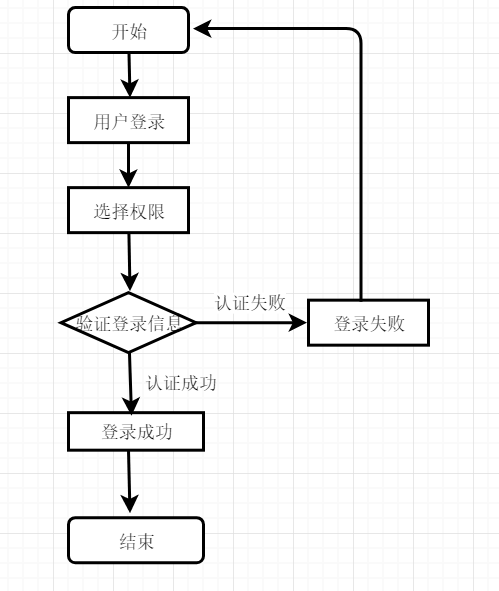


图4-2用户登录程序流程图

### 4.2.1 控制层

登录模块的控制层负责接受来自login.jsp的用户输入，同时调用登入模块的业务逻辑接口，将用户名与密码等用户关键信息传递到业务逻辑层进行判定。等到业务逻辑处理完成之后，将来自业务逻辑层的相应信息传到表现层，并决定显示页面。登入模块控制层列表见表4-3所示。

表 4-3 登入模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 登录 | login.java | SUCCESS | mainFrame.jsp—登入成功，显示提示窗口。 |
| ERROR | login.jsp---登入失败，显示提示窗口。 |

在控制层中的login.java描述如下所示：

**public** **class** login **extends** HttpServlet {}

### 4.2.3 模型层

登录模块的是数据持久层能对用户数据进行增删改查。登入模块数据持久层列表见表4-4所示。

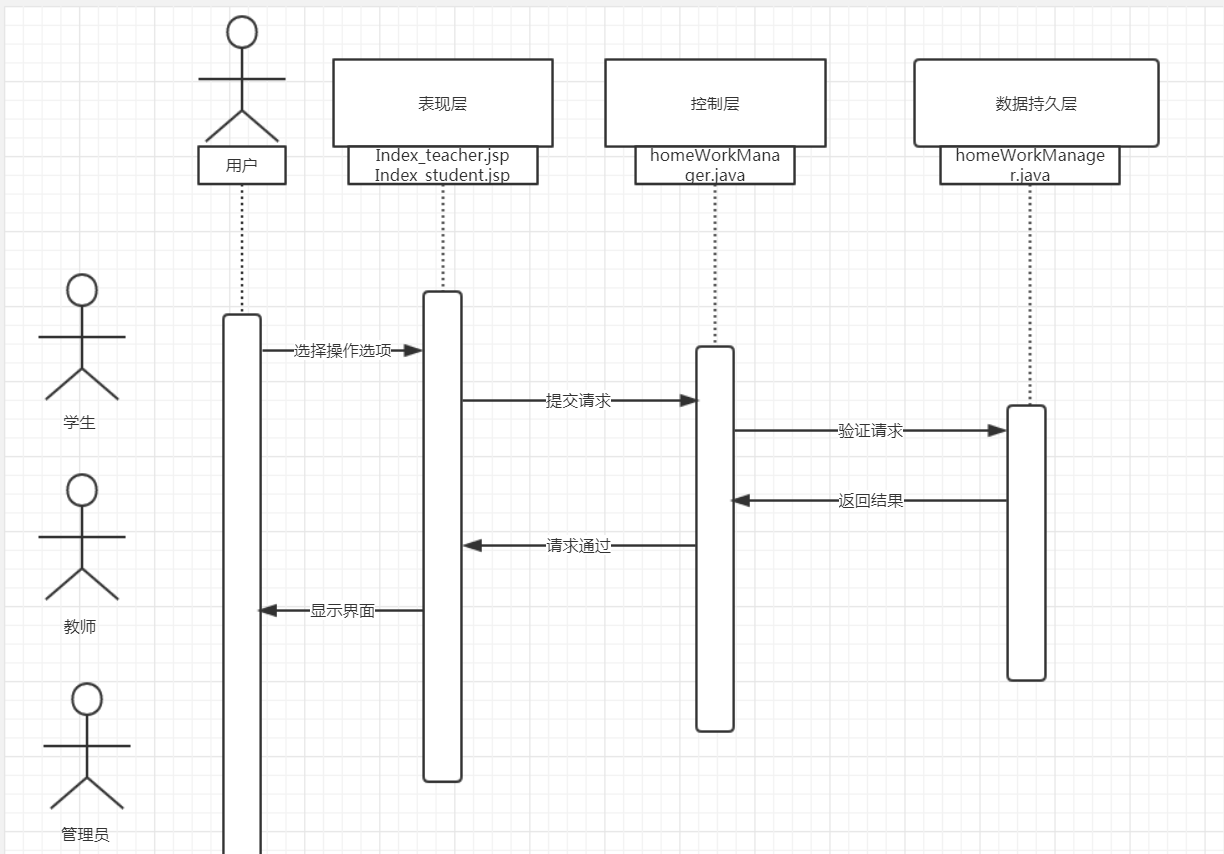
表 4-4 登入模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 登录 | DBbean.java | 调用 user.Java | 对用户信息进行增删改查操作 |

User.java 查操作：

**public** **class** user {}

## 4.3、作业管理模块的详细设计



作业管理模块的系统响应示意图如图4-3所示。

### 4.3.1 表现层

作业管理模块的表现层主要完成对作业发布、提交、批改、查看和下载，由用户选择相应的操作后，跳转到对应页面。对应的jsp页面见表4-7

表4-7

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 教师作业管理页面 | Index\_teacher.jsp | 教师发布、查看、批改、下载作业 |
| 学生作业管理页面 | Index\_student.jsp | 学生完成、查看、下载作业 |

作业管理系统的流程图如图4-4

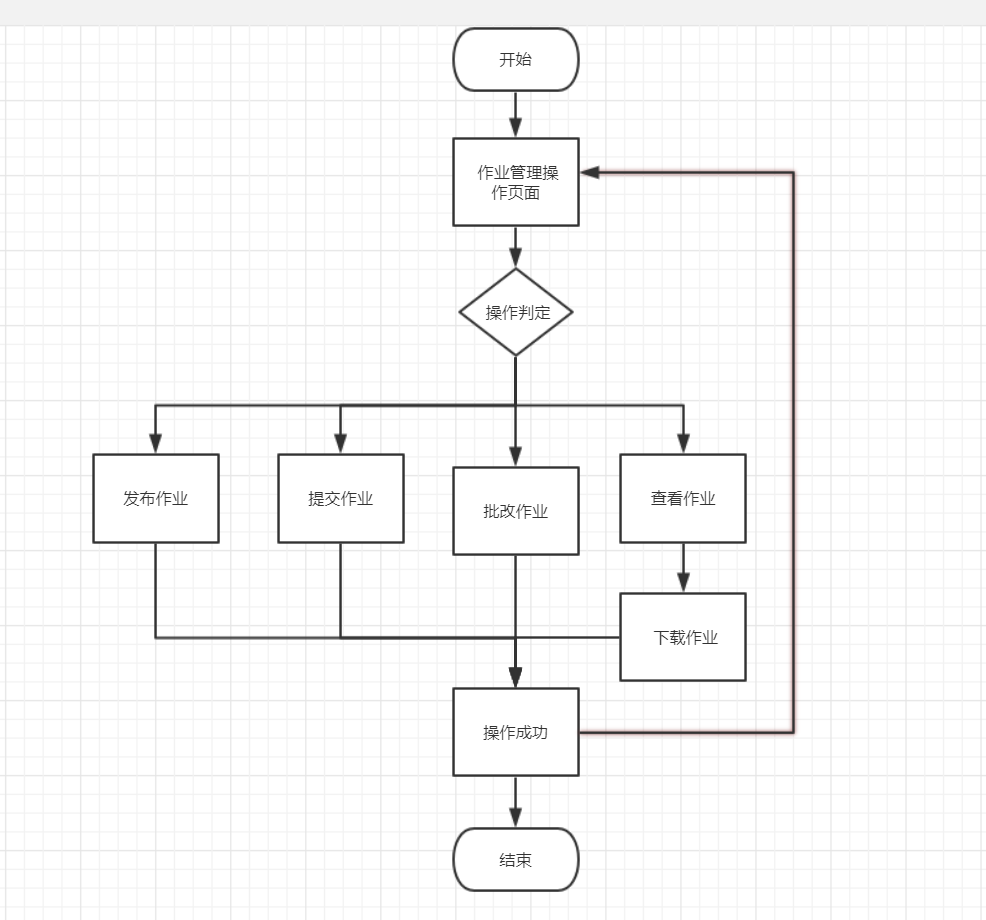


图4-4 作业管理模块流程图

### 4.3.2 控制层

作业管理模块的控制层负责接受来自jobmanage.jsp的用户选择，同时调用作业模块的业务逻辑接口，判定用户选择的操作。等到业务逻辑处理完成之后，将来自业务逻辑层的相应信息传到表现层，并决定显示页面。作业管理模块控制层列表见表4-8所示。

表 4-8 作业管理模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 选择操作 | t\_homework.java | SUCCESS  ERROR | 跳转到选择的窗口 |

t\_homework.java选择操作代码：

**public** **class** t\_homework **extends** HttpServlet {

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

request.setCharacterEncoding("GB2312");

PrintWriter out=response.getWriter();

HttpSession session=request.getSession();

String classid = (String)session.getAttribute("classid");///////////////////////

String submit = request.getParameter("action");

**if**(submit!=**null**&&submit.trim().equals("ac")){

**int** tid =Integer.*parseInt*(request.getParameter("id"));

String title = request.getParameter("title");

System.***out***.print(title);

String file =request.getParameter("file");

String content =request.getParameter("content");

String bdate =request.getParameter("bdate");

String update =request.getParameter("update");

System.***out***.println(title);

homeWorkMamager h = **new** homeWorkMamager();

h.setHtitle(title);

h.setHcontent(content);

h.setBtime(bdate);

h.setCfile(file);

h.setCtime(update);

h.setTid(tid);

h.setClassid(classid);

**if**(h.dcHomework()==1)

{

out.println("很牛逼啊！你的作业发布即将生效！");

**return**;

}**else**

{

out.println("失败！");

}

}

}

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

doGet(request, response);

}

}

### 4.3.3 模型层

作业管理模块的数据持久层主要完成对作业的增删查改。

作业管理模块数据持久层列表见表4-9所示。

表 4-9 作业管理模快数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 登录 | homeWorkManager.java | 调用 homeWorkManager.java | 对作业进行操作 |

homeWorkManager.java查看作业代码：

**public** **class** homeWorkManager {}

# 管理员模块详细设计

管理员模块主要包括添加用户模块、添加课程模块和删除用户模块。添加用户包括添加学生、添加教师、添加班级等功能。

## 5.1、添加用户模块详细设计

添加用户模块的系统响应示意图如图5-1所示

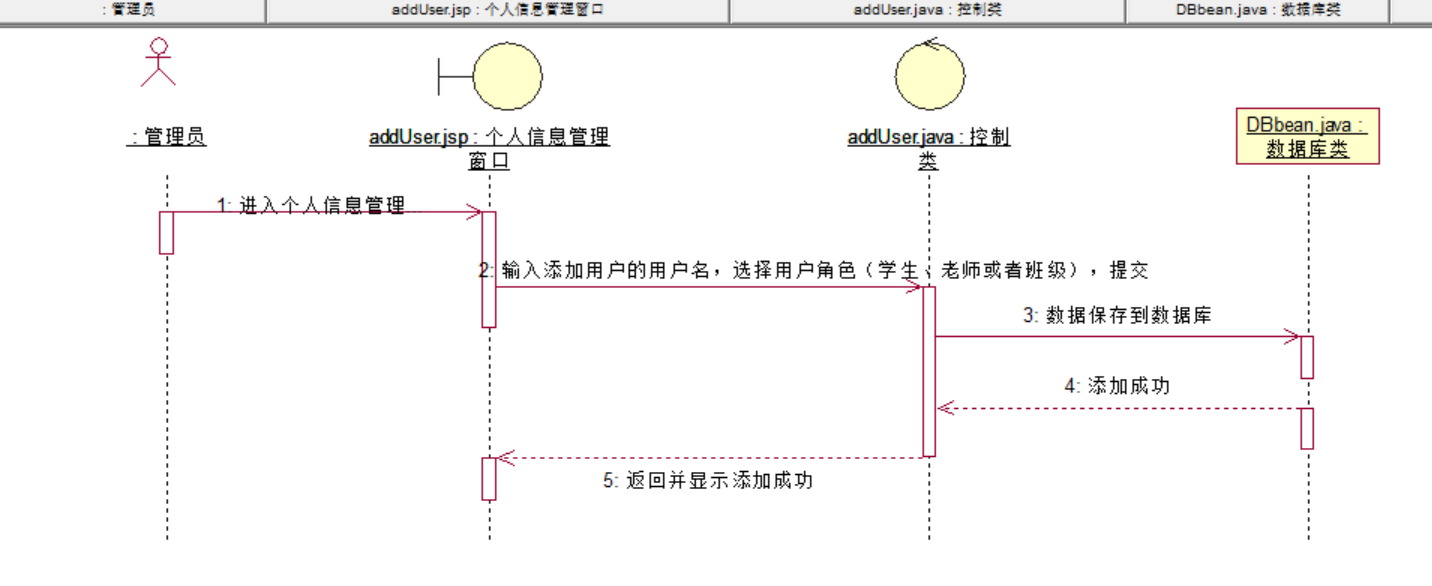


图5-1

### 5.1.1 表现层

添加用户模块表现层主要实现添加学生和老师两个角色和分类，如表5-1-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 添加学生 | addStudent.jsp | 添加学生 |
| 添加教师 | addTeaher.jsp | 添加教师 |
| 添加班级 | addClass.jsp | 添加班级 |

添加用户的流程图如图5-1-1所示

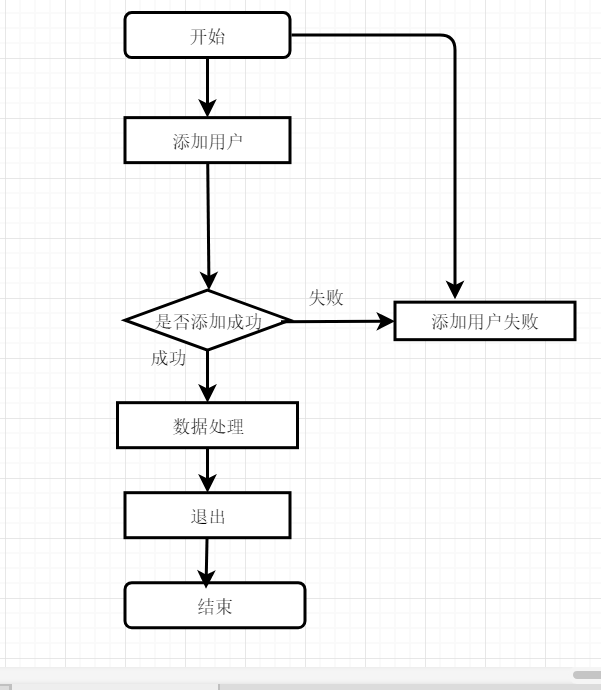


图5-1-1 添加用户流程图

### 5.1.2 控制层

添加用户模块的控制层负责接收添加用户界面（添加学生、添加教师、添加班级）的信息，并向模型层发送添加学生、教师或者是班级的信息，控制层列表见表5-1-2所示

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 添加学生 | addStudent..java | SUCCESS | studentList.jsp |
| ERROR | addStudent..jsp |
| 添加教师 | addTeacher.java | SUCCESS | teacherList.jsp |
| ERROR | addTeacher.jsp |
| 添加班级 | addClass.java | SUCCESS | classList.jsp |
| ERROR | addClass.jsp |

表5-1-2

### 5.1.3 模型层

添加用户模块模型层主要完成对添加用户的判定，同时把数据存到数据库中，模型层列表见表5-1-3。

|  |  |
| --- | --- |
| 模型层 | 描述 |
| DBbean.java | 对用户信息的添加操作 |

表5-1-3

## 5.2、添加课程模块详细设计

添加课程模块的系统响应示意图如图5-2所示

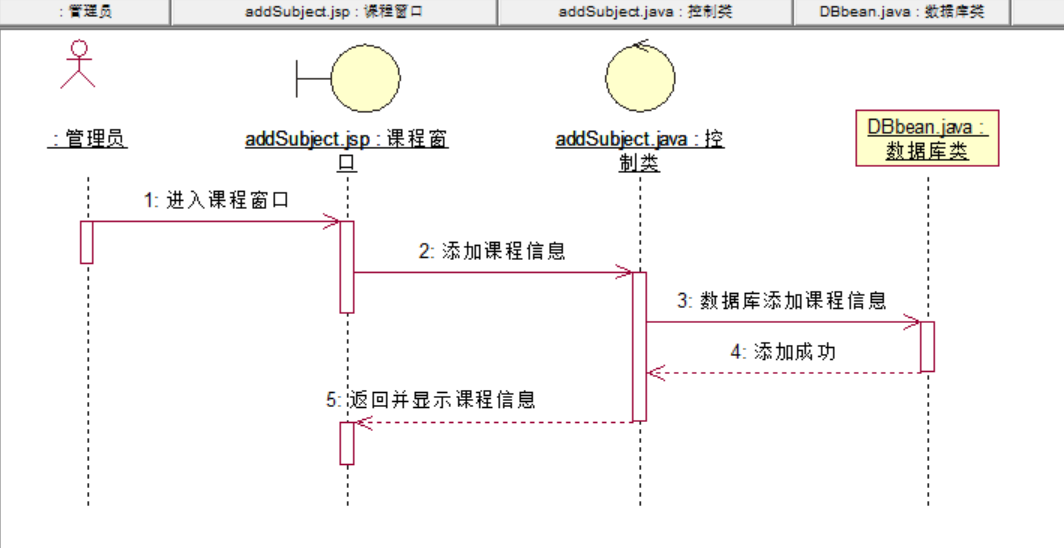


图5-2

### 5.2.1 表现层

添加课程模块主要实现对教学课程的添加，如表5-1-2所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 添加课程 | addSubject.jsp | 添加课程 |

添加课程的流程图如图5-2-1所示

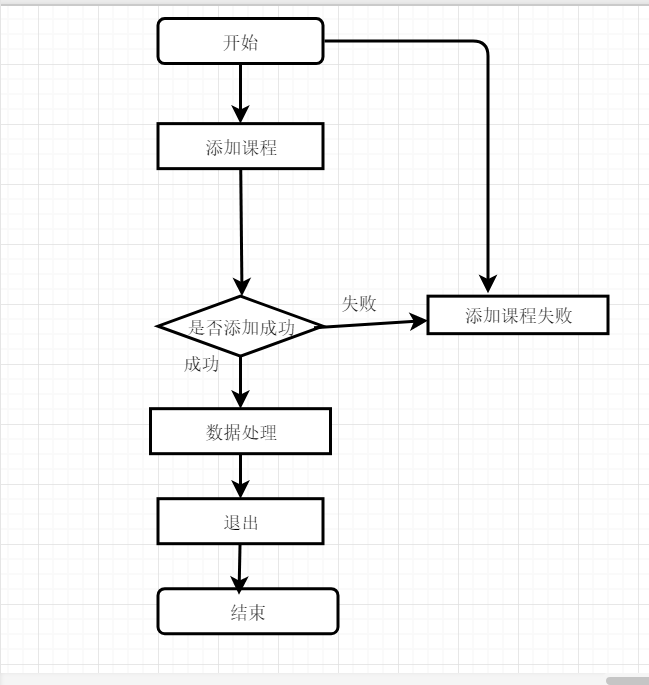


图5-2-1添加课程流程图

### 5.2.2 控制层

添加课程模块控制层主要负责接收添加课程页面的课程信息，并向模型层发送课程信息的数据，控制层列表见表5-2-2所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 添加课程 | addSubject.java | SUCCESS | subjectList.jsp |
| ERROR | addSubject.jsp |

表5-2-2

### 5.2.3 模型层

添加课程模块模型层主要负责判定添加课程的操作，同时把课程信息的数据存在数据库，模型层列表见表5-2-3。

|  |  |
| --- | --- |
| 模型层 | 描述 |
| DBbean.java | 对课程信息的增加操作 |

表5-2-3

## 5.3、删除用户模块详细设计

删除课程模块的系统响应示意图如图5-3所示

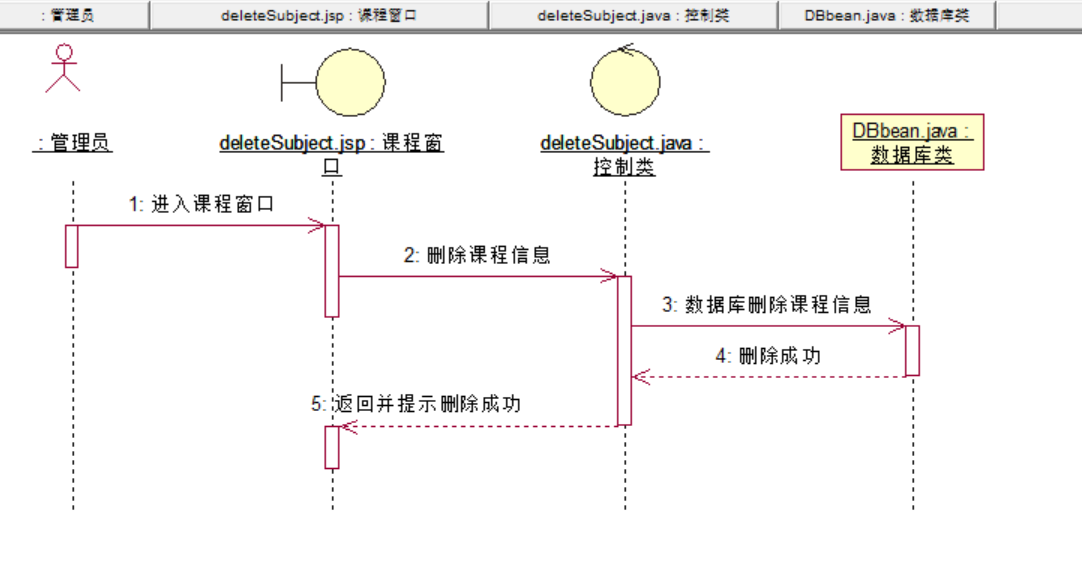


图5-3

### 5.3.1 表现层

删除用户模块主要实现对学生、教师的信息的删除，如表5-3-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 添加学生 | deleteStudent.jsp | 删除学生 |
| 添加教师 | deleteTeaher.jsp | 删除教师 |
| 添加班级 | deleteClass.jsp | 删除班级 |

删除用户的流程图如图5-3-1所示

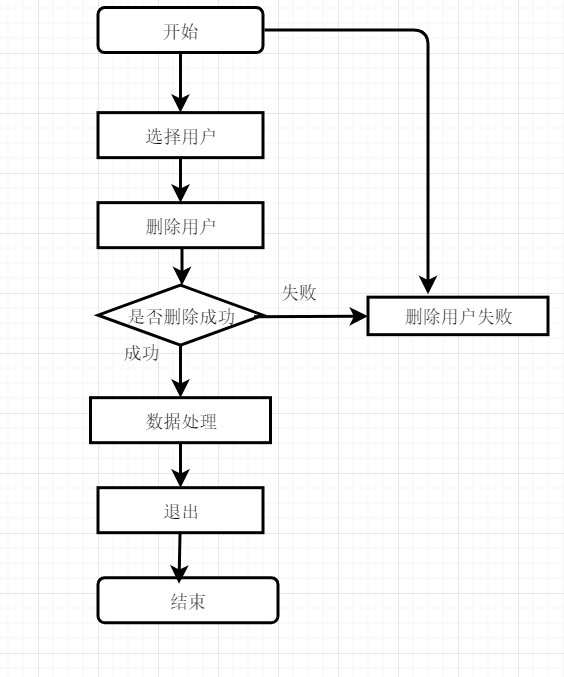


图5-3-1删除用户流程图

### 5.3.2 控制层

删除用户模块控制层主要负责接收删除页面的操作信息，同时向模型层发送删除用户的命令，控制层列表见表5-3-2所示

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 删除学生 | deleteStudent.java | SUCCESS | studentList.jsp |
| ERROR | deleteStudent.jsp |
| 删除教师 | deleteTeacher.java | SUCCESS | teacherList.jsp |
| ERROR | deleteeTeacher.jsp |
| 删除班级 | deleteClass.java | SUCCESS | classList.jsp |
| ERROR | deleteClass.jsp |

图5-3-2

### 5.3.3 模型层

删除用户模块模型层主要负责判定删除课程的操作，同时在数据库删除、更新用户的信息，模型层列表见表5-3-3。

|  |  |
| --- | --- |
| 模型层 | 描述 |
| DBbean.java | 对用户信息的删除操作 |

表5-3-3

# 6、公共部分模块详细设计

## 6.1 查看作业

**6.1.1.描述**

学生用户登录之后需要查看需要做的作业，查看具体的作业要求和相关信息，以便完成作业并提交。

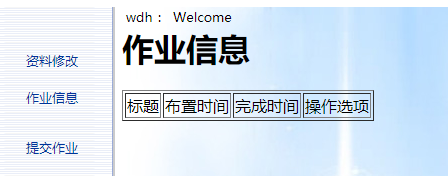
老师登录之后可以查看布置过的作业是否正确。

**6.1.2.功能**

学生：查看本班级已布置的作业。

老师：查看本教师布置的作业。

**6.1.3.参数说明**



从数据库中获得作业信息：作业标题，布置时间，完成时间。点击操作选项可以对作业进行操作，比如做作业。

**6.1.4.文件清单**

|  |  |
| --- | --- |
| 文件名称 | 说明 |
| home\_info.jsp | 显示作业信息的JSP页面 |
| DBbean.java | 连接和查询数据库的工具类 |

## 6.2 登录页面

**6.2.1.描述**

用户进入系统要先输入自己的账号和密码，选择用户角色，以登录系统，根据不同的角色有不同的权限。

**6.2.2.功能**

用户通过输入账号密码和选择角色来验证身份，登录系统。

**6.2.3.参数说明**

**登录：**



账号：用户的账号

密码：用户的登录密码

用户选择：用户选择角色，有学生，老师，管理员。

注册：



参数为用户的基本信息。

**6.2.4.流程逻辑**

注册

输入账号，密码，选择角色

否

账号，密码，角色，是否匹配

是

成功登录

**6.2.5文件清单**

|  |  |
| --- | --- |
| 文件名称 | 说明 |
| login.jsp | 登录的jsp页面 |
| login.java | 登录的servlet |
| index.jsp | 登录后的管理员主页 |
| index\_teacher.jsp | 登陆后的老师主页 |
| index\_student.jsp | 登陆后的学生主页 |
| register.jsp | 注册的jsp页面 |
| register.java | 注册的servlet |

## 6.3 查看个人信息

**6.1.1.描述**

用户登录后可以查看当前账号的详细信息，可以选择修改密码。

**6.1.2.功能**

1查看当前账号的详细信息。

2.可以修改密码。

**6.1.3.参数说明**

详细信息：



可以查看当前用户的姓名，性别，专业，班级。

**6.1.4.流程逻辑**

查看资料

修改密码

输入当前密码

否

当前密码是否正确

是

输入新密码

**6.1.5文件清单**

|  |  |
| --- | --- |
| 文件名称 | 说明 |
| s\_info.jsp | 查看学生信息的jsp文件 |
| user\_fix.jsp | 修改学生账号密码的jsp文件 |
| t\_info.jsp | 查看老师信息的jsp文件 |
| passwordchange.jsp | 修改老师账号密码的jsp文件 |