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

**作业管理系统详细设计**

**V2.1**

**广西民族大学**

评审日期： 2021年6月24日

目录

[1．导言 5](#_Toc75375336)

[1.1 目的 5](#_Toc75375337)

[1.2 范围 6](#_Toc75375338)

[1.3 引用标准 6](#_Toc75375339)

[1.4 参考资料 6](#_Toc75375340)

[1.5 版本更新信息 6](#_Toc75375341)

[2．详细设计简述 7](#_Toc75375342)

[2.1 设计简介 7](#_Toc75375343)

[2.2 模块简介 7](#_Toc75375344)

[3、界面详细设计 8](#_Toc75375345)

[3.1、页面设计说明 8](#_Toc75375346)

[3.2、静态页面详细设计 8](#_Toc75375347)

[4、客户端模块详细设计 9](#_Toc75375348)

[4.1、公用模块的详细设计 9](#_Toc75375349)

[4.1.1 表现层 9](#_Toc75375350)

[4.1.2 控制层 10](#_Toc75375351)

[4.1.3 数据持久层 10](#_Toc75375352)

[4.2、登录模块 13](#_Toc75375353)

[4.2.1 表现层 13](#_Toc75375354)

[4.2.2 控制层 14](#_Toc75375355)

[4.2.3 业务逻辑层 16](#_Toc75375356)

[4.2.4 数据持久层 17](#_Toc75375357)

[4.2.5 域模型层 19](#_Toc75375358)

[4.3、用户修改密码模块 20](#_Toc75375359)

[4.3.1 表现层 21](#_Toc75375360)

[4.3.2 控制层 22](#_Toc75375361)

[4.3.3 业务逻辑层 23](#_Toc75375362)

[4.3.4 数据持久层 23](#_Toc75375363)

[4.3.5 域模型层 24](#_Toc75375364)

[4.4、用户查看个人信息模块 24](#_Toc75375365)

[4.4.1 表现层 25](#_Toc75375366)

[4.4.2 控制层 25](#_Toc75375367)

[4.4.3 业务逻辑层 26](#_Toc75375368)

[4.4.4 数据持久层 27](#_Toc75375369)

[4.4.5 域模型层 27](#_Toc75375370)

[4.5、教师发布作业模块 27](#_Toc75375371)

[4.5.1 表现层 28](#_Toc75375372)

[4.5.2 控制层 31](#_Toc75375373)

[4.5.3 业务逻辑层 31](#_Toc75375374)

[4.5.4 数据持久层 32](#_Toc75375375)

[4.5.5 域模型层 32](#_Toc75375376)

[4.6、教师批改作业及点评模块 32](#_Toc75375377)

[4.6.1 表现层 33](#_Toc75375378)

[4.6.2 控制层 35](#_Toc75375379)

[4.6.3 业务逻辑层 35](#_Toc75375380)

[4.6.4 数据持久层 36](#_Toc75375381)

[4.6.5 域模型层 36](#_Toc75375382)

[4.7、教师选取作业示范模块 37](#_Toc75375383)

[4.7.1 表现层 37](#_Toc75375384)

[4.7.2 控制层 38](#_Toc75375385)

[4.7.3 业务逻辑层 39](#_Toc75375386)

[4.8、学生分数统计模块 39](#_Toc75375387)

[4.8.1 表现层 40](#_Toc75375388)

[4.8.2 控制层 41](#_Toc75375389)

[4.8.3 业务逻辑层 42](#_Toc75375390)

[4.8.4 数据持久层 42](#_Toc75375391)

[4.8.5 域模型层 42](#_Toc75375392)

[4.9、学生提交作业模块 43](#_Toc75375393)

[4.9.1 表现层 43](#_Toc75375394)

[4.9.2 控制层 44](#_Toc75375395)

[4.9.3 业务逻辑层 45](#_Toc75375396)

[4.9.4 数据持久层 45](#_Toc75375397)

[4.9.5 域模型层 45](#_Toc75375398)

[4.10、学生查看批改情况模块 46](#_Toc75375399)

[4.10.1 表现层 47](#_Toc75375400)

[4.10.2 控制层 47](#_Toc75375401)

[4.10.3 业务逻辑层 48](#_Toc75375402)

[4.10.4 数据持久层 48](#_Toc75375403)

[4.10.5 域模型层 49](#_Toc75375404)

[4.11、学生查看作业示范模块 49](#_Toc75375405)

[4.11.1 表现层 50](#_Toc75375406)

[4.11.2 控制层 50](#_Toc75375407)

[4.11.3 业务逻辑层 51](#_Toc75375408)

[4.11.4 数据持久层 51](#_Toc75375409)

[4.11.5 域模型层 52](#_Toc75375410)

[5、管理端模块详细设计 52](#_Toc75375411)

[5.1、用户管理模块详细设计 52](#_Toc75375412)

[5.1.1 表现层 52](#_Toc75375413)

[5.1.2 控制层 53](#_Toc75375414)

[5.1.3 业务逻辑层 53](#_Toc75375415)

[5.1.4 数据持久层 54](#_Toc75375416)

[5.1.5 域模型层 54](#_Toc75375417)

[5.2、课程管理模块详细设计 54](#_Toc75375418)

[5.2.1 表现层 55](#_Toc75375419)

[5.2.2 控制层 55](#_Toc75375420)

[5.2.3 业务逻辑层 55](#_Toc75375421)

[5.2.4 数据持久层 56](#_Toc75375422)

[5.2.5 域模型层 56](#_Toc75375423)

# 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，是反映作业管理系统项目开发项目的工作范围、约束和限制的说明性文件，是进行作业管理系统开发的依据。

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，包括程序描述、输入输出、算法和流程逻辑。

## 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

## 1.4 参考资料

[1] 《软件项目管理案例教程 第3版》 韩万江等 机械工业出版社

## 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 | 操作人 |
| 001 | 2021.6.20 | 0.1 | 全部 | 初始发布版本 | 马润权 |
| 002 | 2021.6.21 | 1.0 | 4章节 | 增加 | 马润权 |
| 003 | 2021.6.21 | 2.0 | 4章节 | 增加、修正 | 马润权、蒙荣毅 |
| 004 | 2021.6.22 | 2.1 | 全部 | 添加代码规范名 | 覃健 |

# 2．详细设计简述

## 2.1 设计简介

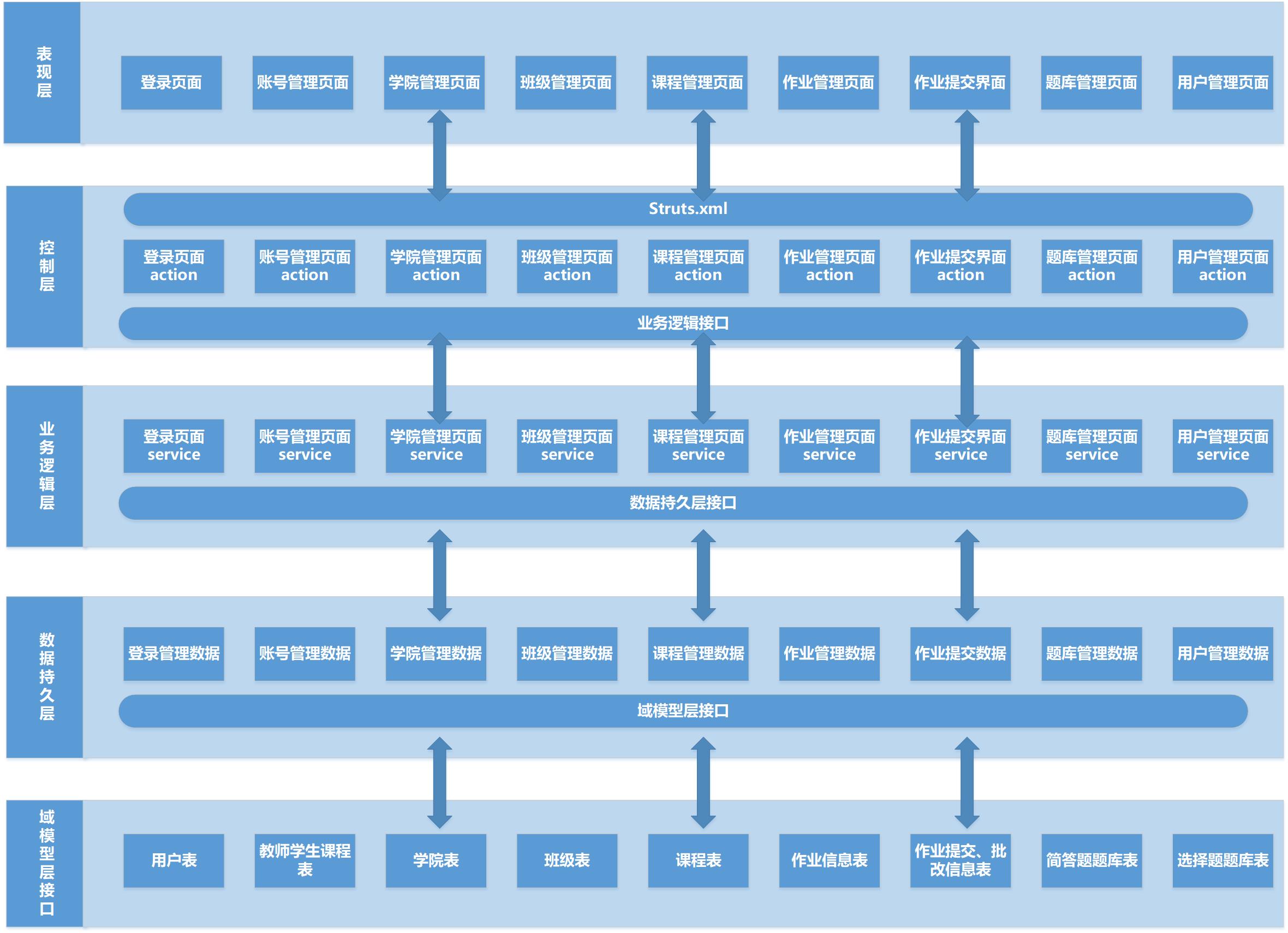
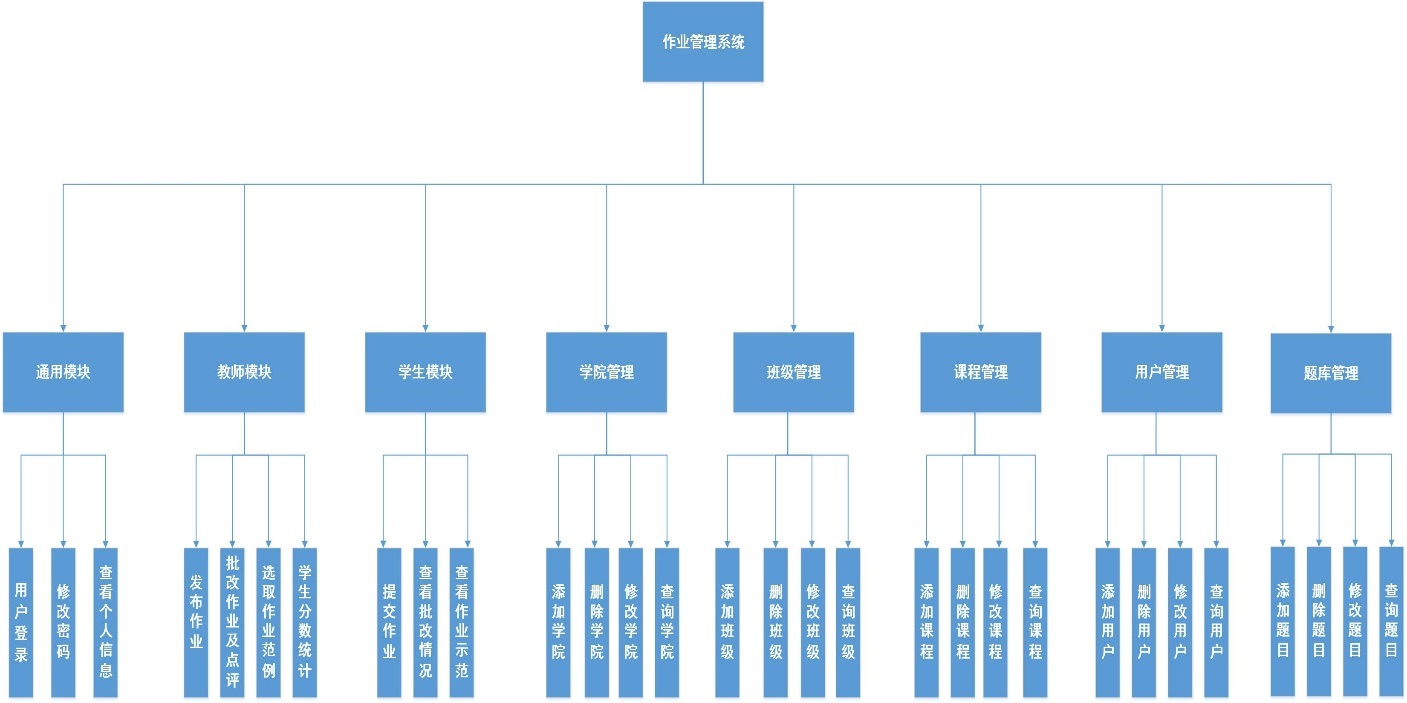
作业管理系统分层结构图如图2-1所示。

图 2-1 作业管理系统分层结构图

## 2.2 模块简介

作业管理系统功能模块图如下图所示。



作业管理系统功能模块图

# 3、界面详细设计

## 3.1、页面设计说明

作业管理系统前端采用jsp+bookstrap+easyUI+jquery技术设计，界面风格美观简洁，易于操作，功能一目了然，有良好的用户体验。

## 3.2、静态页面详细设计

本系统静态页面见表3-1。

表3-1 页面表现层设计

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 登录界面 |  | 验证用户身份进入系统，进入不同的用户界面 |
| 修改密码界面 |  | 用户修改密码 |
| 查看个人信息界面 |  | 查看用户个人详细信息 |
| 发布作业界面 |  | 教师选择题目，设置作业信息（开始时间、截止时间等）后发布作业，所教的学生将收到作业任务 |
| 批改作业及点评界面 |  | 教师批改和点评作业 |
| 提交作业界面 |  | 学生输入答案，提交作业 |
| 查看作业批改情况界面 |  | 查看作业批改情况 |
| 查看作业示范界面 |  | 学生查看教师发布的作业范例 |
| 学院管理界面 |  | 对学院信息进行增删查改 |
| 班级管理界面 |  | 对班级信息进行增删查改 |
| 用户管理界面 |  | 对用户信息进行增删查改 |
| 课程管理界面 |  | 对课程信息进行增删查改 |
| 题库管理界面 |  | 对题库题目进行增删查改 |

# 4、客户端模块详细设计

客户端模块主要包括用户登录、用户修改密码、用户查看个人信息、教师布置作业、教师批改作业及点评、选取作业示范、学生分数统计、提交作业、查看批改情况模块。

## 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，主要是页面出错处理模块。

### 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4-1.1。

表 4-1.1 公用模块表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 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。 |

### 4.1.2 控制层

[公用模块的控制层中需要用到的是各种控制器的父类，例如StudentController.java。此类中实现了很多的实用接口，提供了很多默认方法，这些默认方法包括国际化信息的方法、默认的处理用户请求的方法等，这样可以大大的简化作业管理系统的开发。StudentController.java的部分代码框架()]

package com.ssm.controller;

import com.ssm.pojo.Student;

import com.ssm.service.StudentService;

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

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

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

import java.util.List;

@Controller

@RequestMapping("/Student")

public class StudentController {

@Autowired

@Qualifier("studentServiceImpl")

private StudentService studentService;

//查询全部书籍并展现的jsp

@RequestMapping("/allStudent")

public String listStudent(Model model)

{

List<Student> list = studentService.queryAllStudent();

model.addAttribute("list",list);

return "student/allStudent";

}

}

### 4.1.3 数据持久层

studentMapper.java代码结构：

**package com.ssm.dao;**

**import com.ssm.pojo.Student;**

**import org.apache.ibatis.annotations.Param;**

**import java.util.List;**

**public interface StudentMapper {**

**//添加一个学生**

**int addStudent(Student student);**

**//删除一个学生**

**int deleteStudentById(@Param("studentId") int id);**

**//修改学生信息**

**int updateStudent(Student student);**

**//查询一个学生**

**Student queryStudentById(@Param("studentId")int id);**

**//查询全部学生**

**List<Student> queryAllStudent();**

**//按照名字查询学生**

**List<Student> selectStudentByName(@Param("studentName")String StudentName);**

**}**

studentMapper.xml对studentMapper.java实现：

**<?xml version="1.0" encoding="UTF-8"?>**

**<!DOCTYPE mapper**

**PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"**

**"http://mybatis.org/dtd/mybatis-3-mapper.dtd">**

**<mapper namespace="com.ssm.dao.LoginMapper">**

**<resultMap id="Login" type="com.ssm.pojo.Login">**

**<result property="loginAccount" column="loginAccount"/>**

**<result property="loginPwd" column="loginPwd"/>**

**<result property="loginType" column="loginType"/>**

**</resultMap>**

**<!-- 验证登录-->**

**<select id="checkLogin" parameterType="Login" resultMap="Login">**

**select \***

**from login**

**where loginAccount = #{loginAccount} and loginPwd = #{loginPwd};**

**</select>**

**<!-- 更新类型和密码-->**

**<update id="updateLogin" parameterType="Login">**

**update login**

**set loginPwd=#{loginPwd},loginType=#{loginType}**

**where loginAccount = #{loginAccount};**

**</update>**

**<!--更新密码-->**

**<update id="updateLoginPwd" parameterType="Login">**

**update login**

**set loginPwd=#{newPassword}**

**where loginAccount = #{loginAccount} and loginPwd=#{oldPassword};**

**</update>**

## 4.2、登录模块

在登录模块时，系统内部的相应响应操作示意图如图4-2所示。

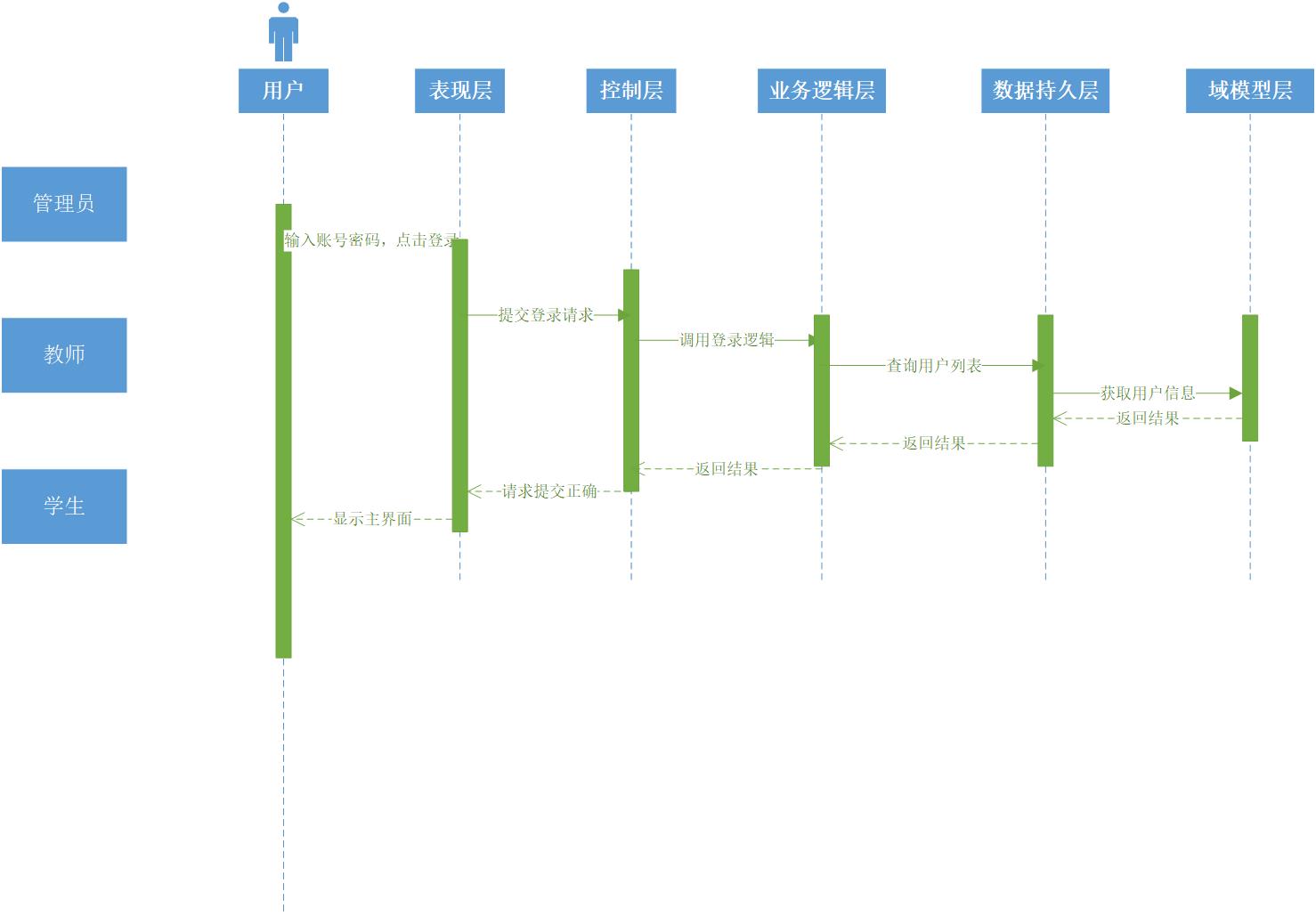


图 4-2 系统响应示意图

### 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4-2.1所示。

表 4-2.1 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（管理员、教师、学生）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。

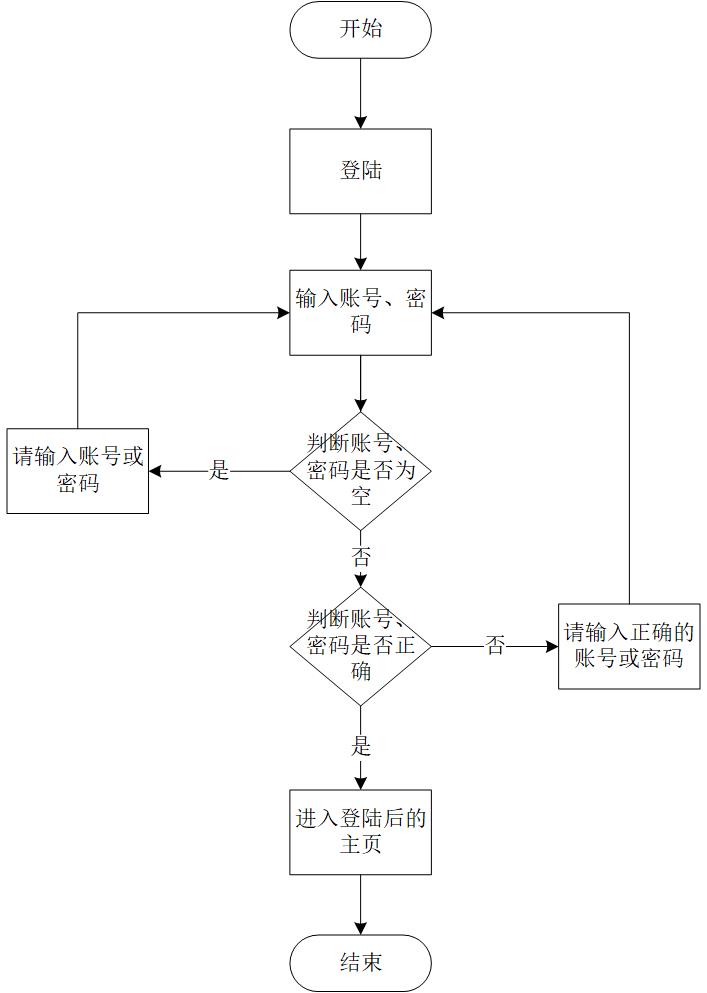
进入游首页

图 4-2.1 登录模块详细设计流程图

### 4.2.2 控制层

登录模块的控制层负责接受来自login.jsp的用户输入，同时调用登录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4-4所示。

表 4-4 登录模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 登录 | LoginController.java  checkUser | 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LoginController.java的描述如下所示：

**@RequestMapping("checkUser")**

**public String check(Login login, ModelMap modelMap){**

**Login login1 = loginService.checkLogin(login);**

**if (login1!=null)**

**{**

**String v = login1.getLoginType();**

**modelMap.addAttribute("Login",login1);**

**switch (v)**

**{**

**case "1":**

**return "teacherIndex" ;**

**case "2":**

**return "studentIndex";**

**case "0":**

**return "adminIndex";**

**}**

**return "../../index";**

**}**

**else**

**{**

**return "../../index";**

**}**

**}**

### 4.2.3 业务逻辑层

登录模块的业务逻辑层主要完成对用户登录逻辑的判定，同时调用登录模块的业务逻辑接口。比如：用户到登录是输入的用户名是否存在、密码是否正确，同时在对用户的身份进行判定。登录模块业务逻辑层列表如图4-2.3所示。

表 4-2.3登入模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 登录 | LoginService.java  LoginServiceImpl.java | LoginMapper | LoginController.java  checkUser |

在登入模块的业务逻辑层是调用了公用的LoginUser接口，同时在实现该接口。

LoginService.java接口主要方法：

**public interface LoginMapper {**

**//验证登录**

**List<Login> selectAllLoginUser();**

**//查看登录表**

**Login checkLogin(Login login);**

**//更新类型和密码**

**Login updateLogin(Login login);**

**//删除**

**int deleteLoginUser(@Param("loginAccount") String loginAccount);**

**//查询用户**

**Login selectByAccount(@Param("loginAccount") String id);**

**//修改密码**

**int updateLoginPwd(@Param("loginAccount")String id,@Param("oldPassword")String oldPassword,@Param("newPassword")String newPassword);**

LoginServiceImpl.java实现了LoginService.java接口， 同时需要调用数据持久层的LoginMapper.Java，利用LoginMapper.xml来实现对数据持久层接口的调用。

LoginServiceImpl.java主要实现属性与方法:

**import com.ssm.dao.LoginMapper;**

**import com.ssm.pojo.Login;**

**import com.ssm.service.LoginService;**

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

**import org.springframework.stereotype.Service;**

**import java.util.List;**

**@Service**

**public class LoginServiceImpl implements LoginService {**

**@Autowired**

**private LoginMapper loginMapper;**

**public List<Login> selectAllLoginUser() {**

**return loginMapper.selectAllLoginUser();**

**}**

**public Login checkLogin(Login login) {**

**return loginMapper.checkLogin(login);**

**}**

**public Login updateLogin(Login login) {**

**return loginMapper.updateLogin(login);**

**}**

**public int deleteLoginUser(String loginAccount) {**

**return loginMapper.deleteLoginUser(loginAccount);**

**}**

**}**

### 4.2.4 数据持久层

修改登录信息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4-2.4所示。

表 4-2.4 登入模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 修改登录信息 | LoginMapper.java  LoginMapper.xml | 调用 Login.Java | 对用户信息进行增删改查操作 |

Login.java定义了用户信息进行增、删、改、查的接口：

public interface LoginMapper {

//验证登录

List<Login> selectAllLoginUser();

//查看登录表

Login checkLogin(Login login);

//更新类型和密码

Login updateLogin(Login login);

//删除

int deleteLoginUser(@Param("loginAccount") String loginAccount);

//查询用户

Login selectByAccount(@Param("loginAccount") String id);

//修改密码

int updateLoginPwd(@Param("loginAccount")String id,@Param("oldPassword")String oldPassword,@Param("newPassword")String newPassword);

LoginMapper.xml是LoginMapper.java接口的实现：

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE mapper

PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"

"http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="com.ssm.dao.LoginMapper">

<resultMap id="Login" type="com.ssm.pojo.Login">

<result property="loginAccount" column="loginAccount"/>

<result property="loginPwd" column="loginPwd"/>

<result property="loginType" column="loginType"/>

<!-- 更新类型和密码-->

<update id="updateLogin" parameterType="Login">

update login

set loginPwd=#{loginPwd},loginType=#{loginType}

where loginAccount = #{loginAccount};

</update>

</mapper>

### 4.2.5 域模型层

登录模块用到域模型层中Login.java，Login.java是一个公用域模型，在涉及到用户信息查询等操作时，就会调用到该模型，登录模块域模型层列表如表4-7所示。

表 4-2.5登入模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| Login.java | 登录或修改密码需要操作该实体类 |

Login.java主要属性与方法：

使用了lombok组件，可以帮助我们自动写get和set方法，无参和有参的构造方法

**package com.ssm.pojo;**

**import lombok.AllArgsConstructor;**

**import lombok.Data;**

**import lombok.NoArgsConstructor;**

**@Data**

**@AllArgsConstructor**

**@NoArgsConstructor**

**public class Login {**

**private String loginAccount;**

**private String loginPwd;**

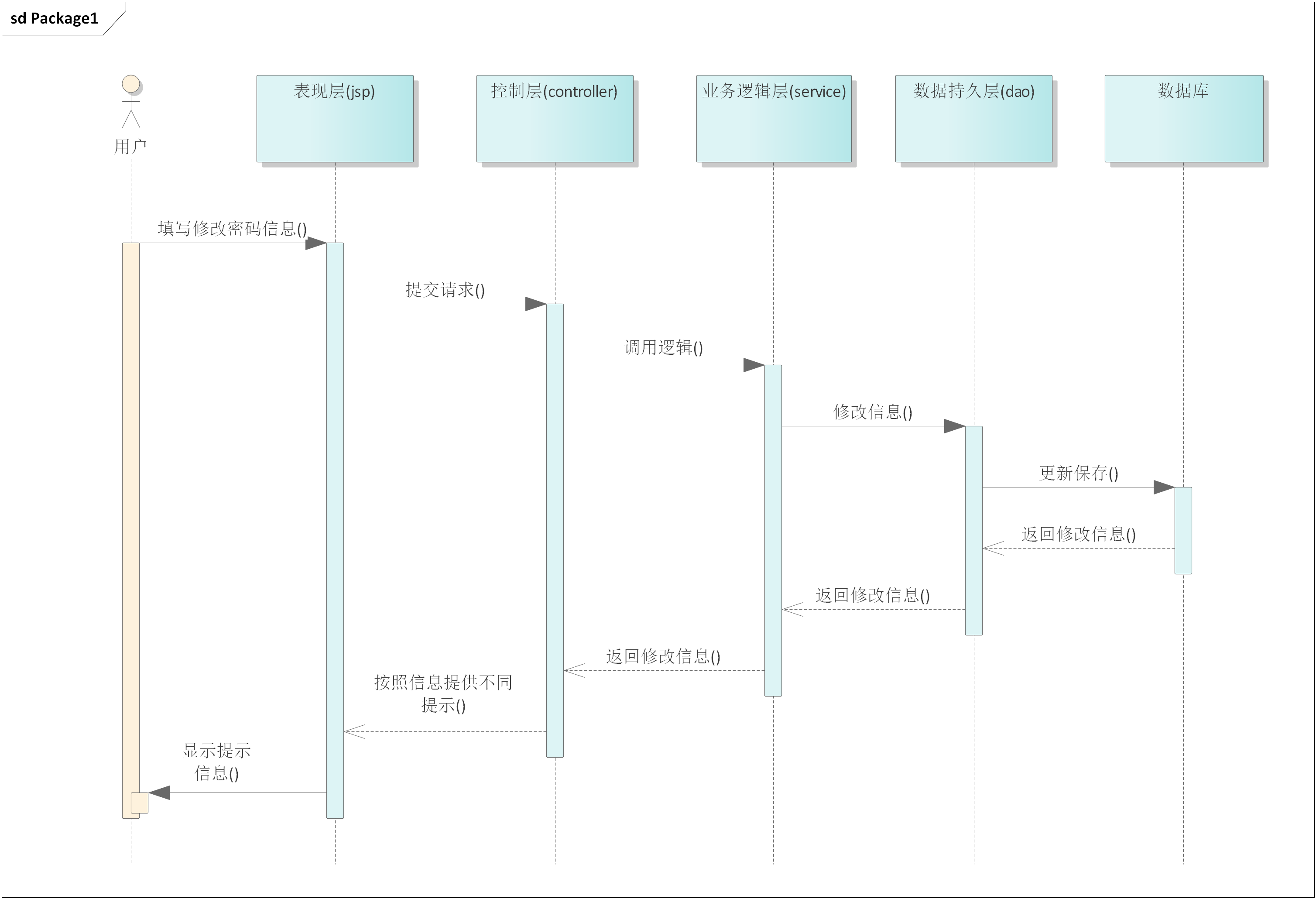
**private String loginType;**

**}**

图片：4-2-1

## 4.3、用户修改密码模块

用户修改密码模块的系统内部的相应响应操作示意图如图4-3所示。



4-3 用户修改密码响应示意图

### 4.3.1 表现层

用户修改密码模块的表现层主要是完成不同用户的密码修改功能，用户输入旧密码，再输入两次新密码，确认后页面给出响应消息，提示密码修改成功或错误提示。表现层对应的jsp页面列表见表4-3.1所示。

表 4-3.1 用户修改密码模块表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | jsp | 功能描述 |
| 管理员,老师 | update\_password.jsp | 教师、管理员、学生修改密码 |

修改密码的流程图如图4-3.1所示：



图 4-3 修改密码模块详细设计流程图

### 4.3.2 控制层

修改个人密码模块的控制层负责接受来自.jsp的用户输入，同时调用修改个人密码模块的业务逻辑接口，将新密码等信息传递给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给表现层，并显示给页面。修改个人密码模块的控制层列表见表4-3.2所示。

表 4-3.2 修改个人密码模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 | 转移说明 | 出口 |
| 修改密码 | LoginController.java  UpdatePassword | SUCCESS | update\_password.jsp，提示修改成功 |
| ERROR | update\_password.jsp提示修改失败 |

在控制层中LoginController.java的描述如下所示：

### 4.3.3 业务逻辑层

修改个人密码模块的业务逻辑层主要完成用户修改密码逻辑的判定，同时调用修改密码模块的业务逻辑接口。比如：修改个人密码模块业务逻辑层列表如表4-3.3所示。

表 4-3.3修改个人密码模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 修改个人密码 | LoginService.java | 调用LoginMapper | 返回controller |

在修改个人密码模块的业务逻辑层是调用了公用的LoginService接口，同时在实现该接口。

LoginService.java接口的修改密码主要方法：

String updateLoginPwd(String id,String oldPassword,String newPassword);

LoginServiceImpl.java主要实现属性与方法:

public String updateLoginPwd(String id, String oldPassword, String password) {

String judge="0";

Login login=loginMapper.selectByAccount(id);

if (login!=null)

{

if (login.getLoginPwd().equals(oldPassword)){

loginMapper.updateLoginPwd(id,oldPassword,password);

judge="1";//更新成功

System.out.println("更新成功");

}

else {

judge="2";//更新失败

System.out.println("更新成功");

}

}

return judge;

}

### 4.3.4 数据持久层

用户修改个人密码模块的是数据持久层能对用户数据进行增删改查。用户修改个人模块数据持久层列表见表4-3.4所示。

表 4-3.4 修改个人密码模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 用户修改密码 | LoginMapper.java | 调用 Login.java | 对用户信息进行修改操作 |

LoginMapper.xml定义了用户对密码进行修改的实现：

<update id="updateLoginPwd" parameterType="Login">

update login

set loginPwd=#{newPassword}

where loginAccount = #{loginAccount} and loginPwd=#{oldPassword};

</update>

### 4.3.5 域模型层

修改个人密码模块用到域模型层中Login.java，Login.java是一个公用域模型，在涉及到用户修改密码等操作时，就会调用到该模型。

表 4-3.5修改个人密码模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| Login.java | 存储用户的信息 |

Login.java主要属性与方法：详情查看图片：4-2-1

## 4.4、用户查看个人信息模块

用户查看个人信息模块的系统内部的相应响应操作示意图如图4-4所示。

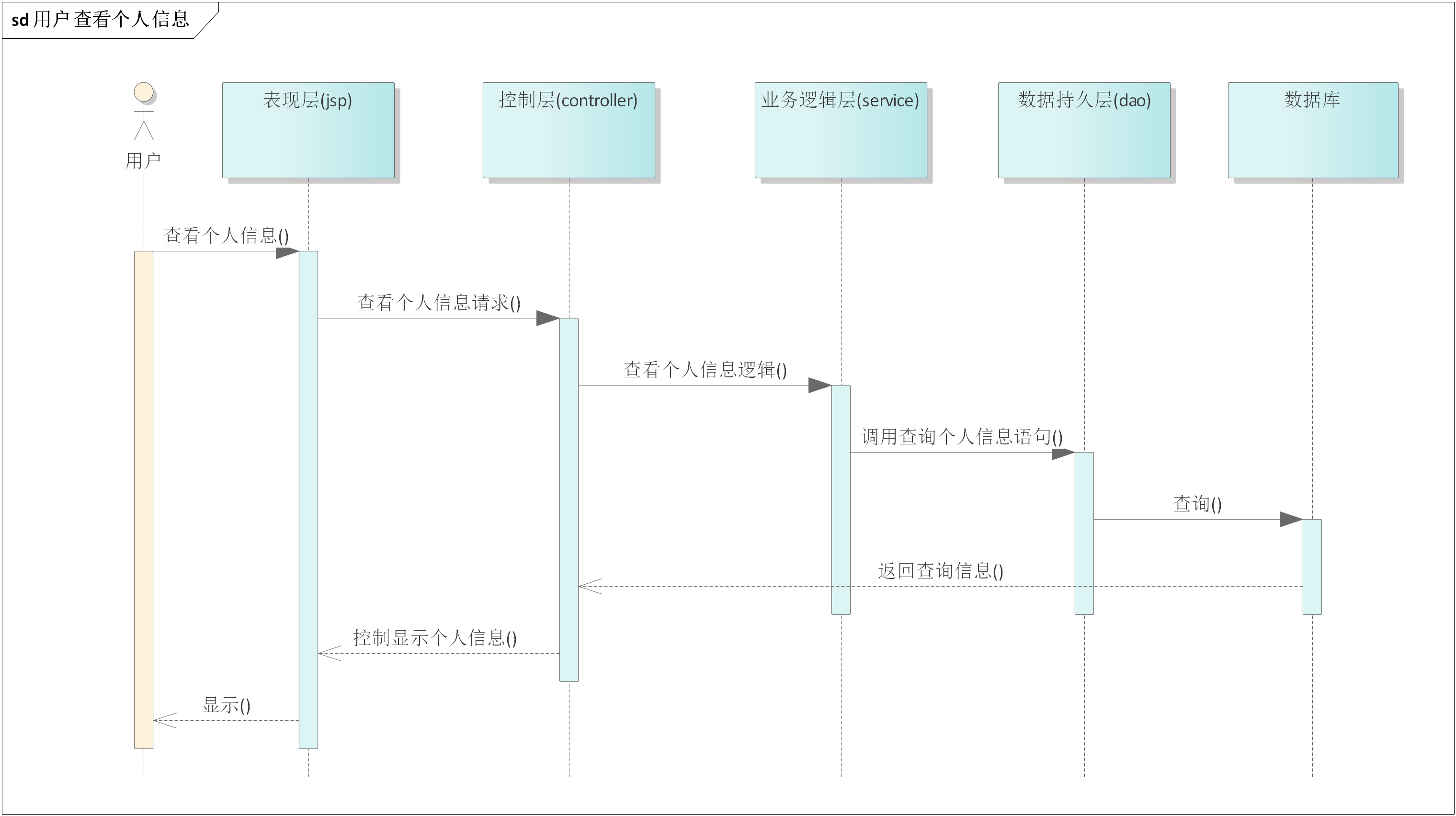


图 4-4 用户查看个人信息响应示意图

### 4.4.1 表现层

用户查看个人信息模块的表现层主要用于查看用户个人信息，确认后页面给出响应消息，显示用户个人信息。表现层对应的jsp页面列表见表4-1.1所示。

表 4-4.1 用户查看个人信息模块表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | jsp | 功能描述 |
| 管理员个人信息页面 | AmdinSelf.jsp | 管理员查看自己的信息 |
| 教师个人信息页面 | teacherSelf.jsp | 教师查看自己的信息 |
| 学生个人信息页面 | studentSelf.jsp | 学生查看自己的信息 |

用户查看个人信息的流程图如图4-4.1所示：



图 4-4.1 用户查看个人信息模块详细设计流程图

### 4.4.2 控制层

用户查看个人信息模块的控制层负责接受来自AmdinSelf.jsp，teacherSelf.jsp studentSelf.jsp的查看个人信息请求，同时调用查看个人信息模块的业务逻辑接口，将请求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。用户查看个人信息模块控制层列表见表4-4.2所示。

表 4-4.2 用户查看个人信息模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 | 转移说明 | 出口 |
| 查看个人信息 | MyInfo.java | 跳转到AmdinSelf.jsp, teacherSelf.jsp, studentSelf.jsp | |

在控制层MyInfo.java功能描述如下所示

### 4.4.3 业务逻辑层

用户查看个人信息模块的业务逻辑层主要完成对用户查看个人信息逻辑的判定，同时调用用户查看个人信息模块的业务逻辑接口。用户查看个人信息模块业务逻辑层列表如表4-4.3所示。

表 4-4.3登陆模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 查看个人信息 | StudentService.java  teacherService.java  adminService.java | studentMapper.java  teacherMapper.java  adminMapper.java | 返回给Api |

在查看个人信息模块的业务逻辑层是调用了StudentService，teacherService，adminService接口，同时在实现该接口。

studentService接口主要方法：getMyInfo(String id)

teacherService接口主要方法: getMyInfo(String id)

adminService接口主要方法:getMyInfo(String id)

studentServiceimpl.java,teacherServiceimpl.java, adminServiceimpl.java

实现了studentService. java,teacherService. java,adminService.java接口， 同时需要调用数据持久层的studentMapper.xml,teacherMapper.xml,adminMapper.xml，实现对数据持久层接口的调用。

### 4.4.4 数据持久层

用户查看个人信息模块的数据持久层能对用户数据进行增删改查。用户查看个人信息模块数据持久层列表见表4-4.4所示。

表 4-4.4 用户查看个人信息模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 查看用户自己的信息 | studentMapper.xml  teacherMapper.xml  adminMapper.xml | student.java  teacher.java  admin.java | 由xml文件里的SQL语句对用户的信息进行查询 |

studentMapper.xml

teacherMapper.xml

adminMapper.xml定义了用户信息查询的接口：

### 4.4.5 域模型层

用户查看个人信息模块用到域模型层中student.java，teacher.java，admin.java用户信息查询等操作时，就会调用到该模型，用户查看个人信息模块域模型层列表下表所示。

表 4-1.5用户查看个人信息模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| student.java | 学生实体类 |
| teacher.java | 老师实体类 |
| admin.java | 管理员 |

## 4.5、教师发布作业模块

教师发布作业模块的系统内部的相应响应操作示意图如图4-5所示。

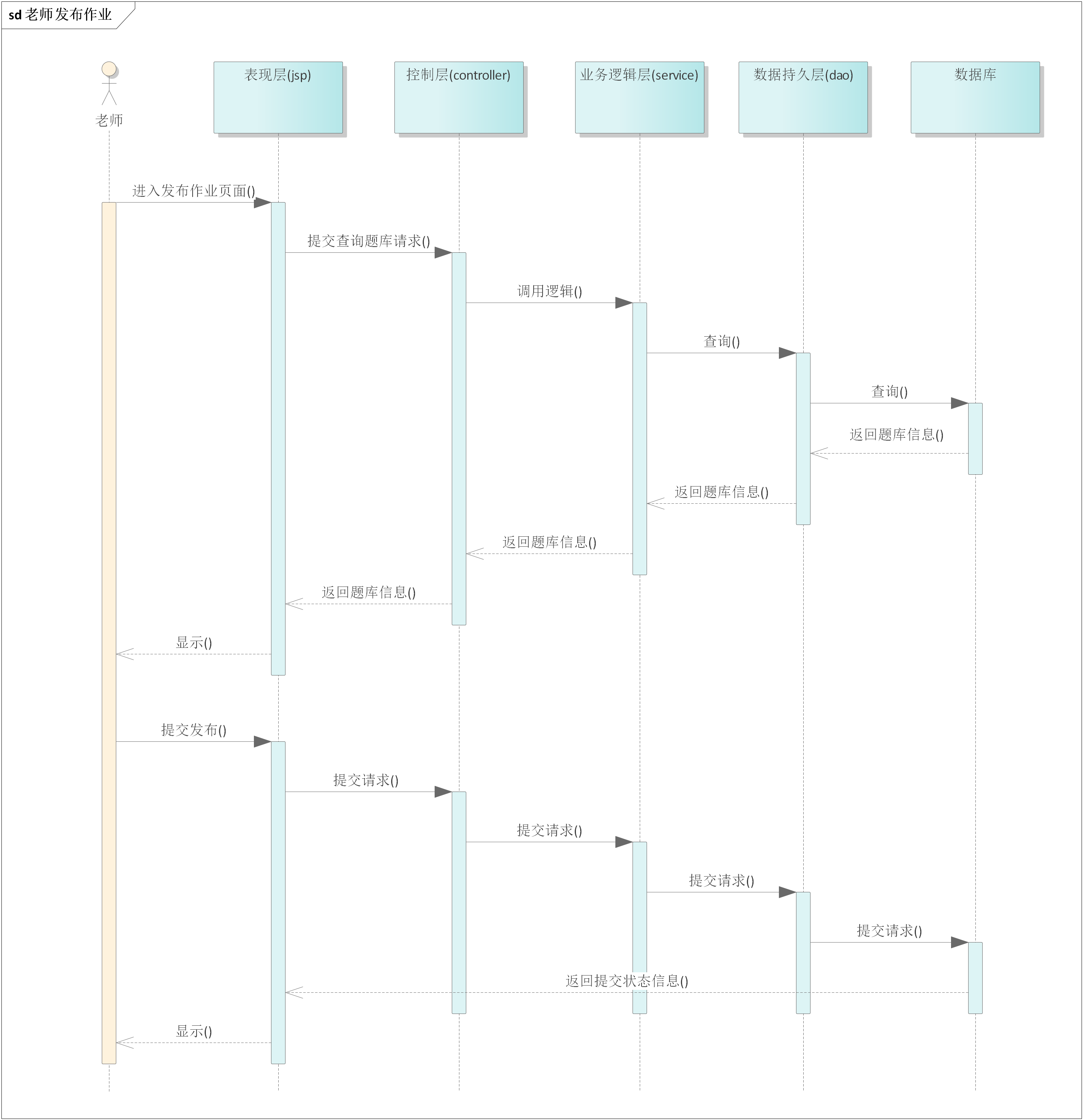


图 4-5 教师发布作业模块的信息响应示意图

### 4.5.1 表现层

教师发布作业模块的表现层主要完成发布作业的功能，页面要求教师输入作业信息，从题库中选择题目，确认发布后页面给出响应消息，提示发布成功或失败的提示。表现层对应的jsp页面列表见表4-5.1所示。

表 4-5.1 教师发布作业模块表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | jsp | 功能描述 |
| 老师课程页面 | myCourse.jsp | 显示老师所有的课程 |
| 老师发布作业界面 | addHomework.jsp | 添加学生页面 |

教师发布作业的流程图如图4-5.1所示：



图 4-5.1 教师发布作业模块详细设计流程图

### 4.5.2 控制层

教师发布作业模块的控制层负责接受来自myCourse.jsp、addHomework.jsp的教师输入，同时调用教师发布作业模块的业务逻辑接口，将作业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。教师发布作业模块控制层列表见表4-5.2所示。

表 4-5.2 教师发布作业模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 |  | 出口 |
| 查看课程 | TCSController.java  selectMyCourse | 查询到课程后跳转myCourse.jsp界面 | |
| 布置作业 | HomeworkController.java  addSAQ | 发布作业返回 | |

在控制层TCSController. selectMyCourse, HomeworkController. addSAQ功能描述如下所示

### 4.5.3 业务逻辑层

教师发布作业模块的业务逻辑层主要完成对教师发布作业逻辑的判定，同时调用教师发布作业模块的业务逻辑接口。比如：作业必填信息是否填写完整，同时再对教师提交的作业信息进行判定。教师发布作业模块业务逻辑层列表如表4-5.3所示。

表 4-5.3教师发布作业模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 老师发布作业 | HomeworkService.java  TCSService.java | HomeworkMapper  TCSMapper | 返回给Api |

在教师发布作业模块的业务逻辑层是调用了公用的TCSService, HomeworkService接口，同时在实现该接口。

HomeworkService.java接口主要方法：int insertHomework(HomeworkInfo homeworkInfo);

TCSService: List<Course> selectMyCourse(String teacherAccount);

HomeworkServiceImpl.java, TCSServiceimpl.java实现了HomeworkService.java, TCSService.java接口， 同时需要调用数据持久层的HomeworkMapper.xml,TCSMapper.xml，实现对数据持久层接口的调用。

### 4.5.4 数据持久层

教师发布作业模块是数据持久层能对作业数据进行增删改查。教师发布作业模块数据持久层列表见表4-5.4所示。

表 4-5.4 教师发布作业模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 老师发布作业 | HomeworkMapper.java | HomeworkInfo.java | 对用户信息进行增删改查操作 |

SysUserMapper.java定义了用户信息进行增、删、改、查的接口：

public interface HomeworkMapper {  
 //老师添加作业  
 int insertHomework(HomeworkInfo homeworkInfo);  
 //老师查看正在做的作业  
 List<HomeworkInfo> NewHomework(@Param("NowTime") String NowTime, @Param("teacherAccount") String teacherAccount, @Param("courseId") Integer courseId);  
 //学生获取作业  
 List<ShortAnswerQuestions>getMyHomework(@Param("courseId") Integer courseId,@Param("nowTime") String NowTime);  
 //学生写简答题  
 int doShortHomework(@Param("questionAnswer")String questionAnswer,@Param("studentAccount") String studentAccount,@Param("courseId") Integer courseId,@Param("nowTime") String nowTime,@Param("saqId") Integer saqId);  
}

### 4.5.5 域模型层

教师发布作业模块用到的域模型层中HomeworkInfo.java，和TCS.java，他们如表4-5.5所示。

表 4-5.5教师发布作业模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| HomeworkInfo.java | 老师发布作业的信息 |
| TCS | 老师课程学生关联表 |

## 4.6、教师批改作业及点评模块

教师批改作业及点评模块的系统内部的相应响应操作示意图如图4-6所示。

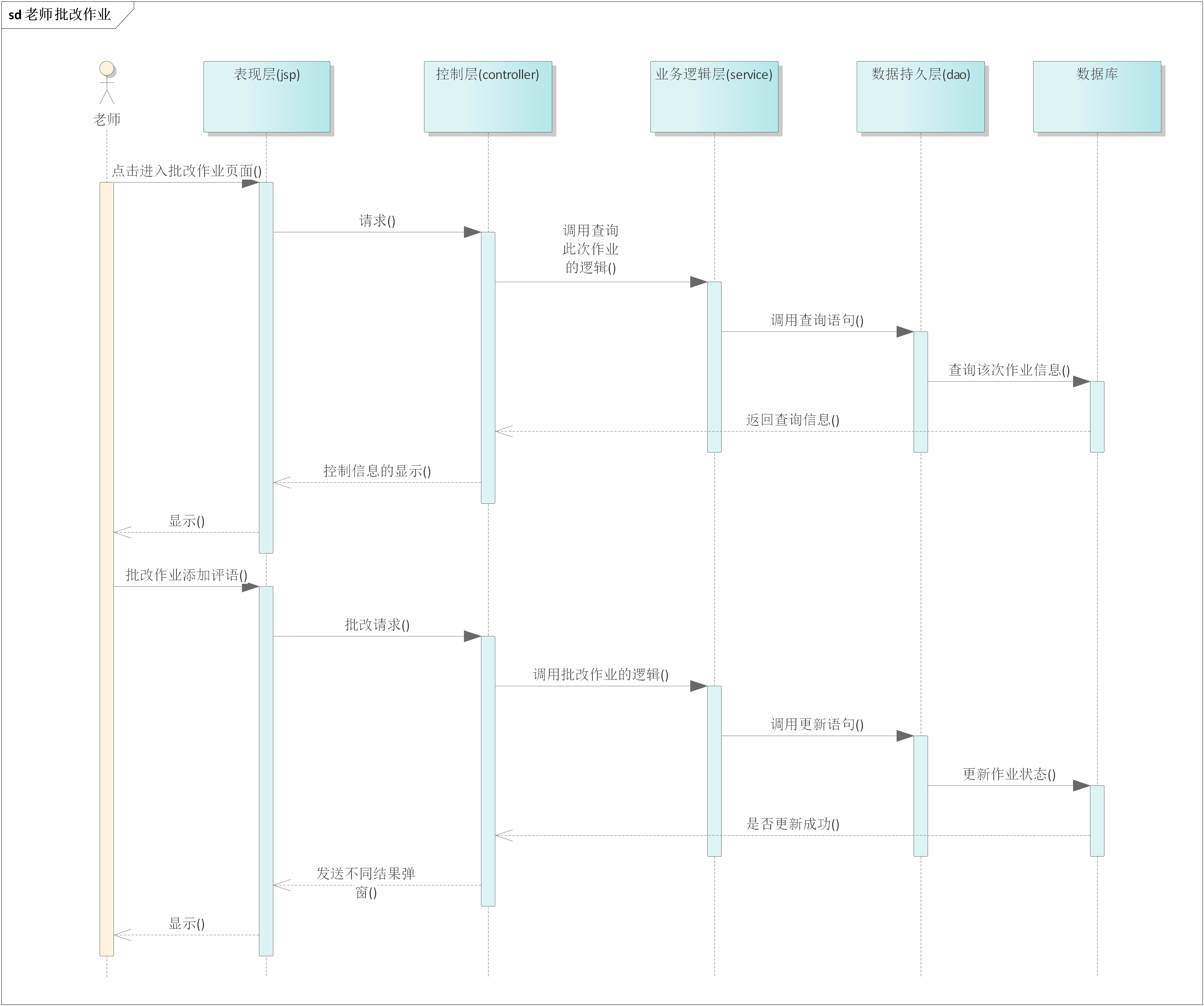


图 4-6 教师批改作业及点评模块的信息响应示意图

### 4.6.1 表现层

教师批改作业及点评模块的表现层主要完成作业的批改和点评功能，批改作业及点评页面要求教师对学生作业进行评分和点评，确认后页面给出响应消息，提示批改成功或失败的提示。表现层对应的jsp页面列表见表4-6.1所示。

表 4-6.1 教师批改作业及点评模块表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | jsp | 功能描述 |
| 教师批改作业 | teacherCorrection.jsp | 教师批改作业的界面 |

教师批改作业及点评的流程图如图4-6.1所示：



图 4-6.1 教师批改作业及点评模块详细设计流程图

### 4.6.2 控制层

教师批改作业及点评模块的控制层负责接受来自login.jsp、pcweblogin.jsp的教师批改和点评信息情况，同时调用教师批改作业及点评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。教师批改作业及点评模块控制层列表见表4-6.2所示。

表 4-6.2 教师批改作业及点评模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 |  | 出口 |
| 教师批改作业 | HomeworkRecod. Correction | 老师批改作业成功 | |

在控制层HomeworkRecod. Correction功能描述如下所示

### 4.6.3 业务逻辑层

教师批改作业及点评模块的业务逻辑层主要完成对教师批改作业及点评逻辑的判定，同时调用教师批改作业及点评模块的业务逻辑接口。登陆模块业务逻辑层列表如表4-1.3所示。

表 4-2.3教师批改作业及点评模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 教师批改作业 | HomeworkRecodService | 调用HomeworkRecodMapper | 返回给Api |

在教师批改作业及点评模块的业务逻辑层是调用HomeworkRecodService接口，同时在实现该接口。

HomeworkRecodService.java接口主要方法：

HomeworkRecodServiceImpl.java实现了HomeworkRecodService.java接口， 同时需要调用数据持久层的HomeworkRecodMapper.Java，实现对数据持久层接口的调用。

### 4.6.4 数据持久层

教师批改作业及点评模块的数据持久层能对作业数据进行增删改查。教师批改作业及点评模块数据持久层列表见表4-1.4所示。

表 4-1.4 教师批改作业及点评模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 老师批改作业 | HomeworkRecodService.java | 调用 HomeworkRecod.java | 对学生作业进行修改查询等 |

HomeworkRecodService.java定义了作业信息改、查的接口：

### 4.6.5 域模型层

教师批改作业及点评模块用到域模型层中HomeworkRecod.java，HomeworkRecod.java是一个模型，在涉及到学生作业查询等操作时，就会调用到该模型，登陆模块域模型层列表如表4-6.5所示。

表 4-6.5教师批改作业及点评模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| HomeworkRecod.java | 学生作业的记录模型 |

HomeworkRecod.java主要属性与方法：

public class HomeWorkRecord {

private int hrId;

private Integer courseId;

private String teacherAccount;

private String questionType;

private int questionId;

private String questionAnswer;

private String studentAccount;

private String correctionState;

private float achievement;

private String homeworkStartTime;

private String homeworkEndTime;

## 4.7、教师选取作业示范模块

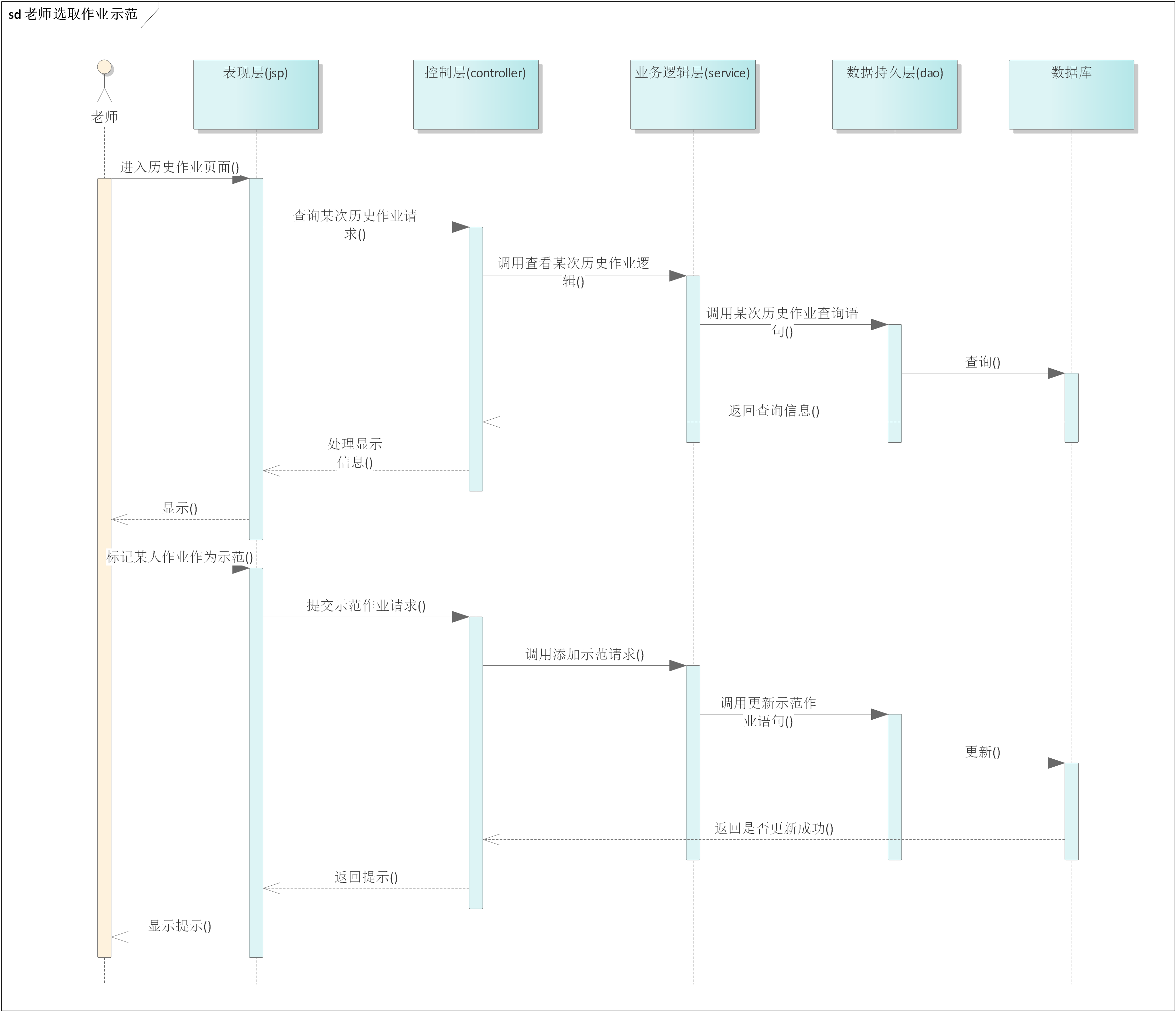
教师选取作业示范模块的系统内部的相应响应操作示意图如图4-7所示。

图 4-7 教师选取作业示范模块的信息响应示意图

### 4.7.1 表现层

教师选取作业示范模块的表现层主要完成优秀作业范例选取的功能，教师选取作业示范页面要求教师选择某学生的作业作为范例，确认后页面给出响应消息，提示提交成功或失败的提示。表现层对应的jsp页面列表见表4-7.1所示。

表 4-7.1 教师选取作业示范模块表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | jsp | 功能描述 |
| 教师选取优秀登录界面 | excellentHomework.jsp | 选择优秀的作业 |

教师选取作业示范的流程图如图4-7.1所示：



图 4-7 教师选取作业示范模块详细设计流程图

### 4.7.2 控制层

教师选取作业示范模块的控制层负责接受来自excellentHomework.jsp的作业范例，同时调用教师选取作业示范模块的业务逻辑接口，将作业范例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。登陆模块控制层列表见表4-7.2所示。

表 4-7.2 教师选取作业示范模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 |  | 出口 |
| 选取示范作业 | HomeworkRecod.get excellentHomework | 添加示范作业成功 | |

在控制层HomeworkRecod.get excellentHomework功能描述如下所示

### 4.7.3 业务逻辑层

教师选取作业示范模块的业务逻辑层主要完成对教师选取作业示范逻辑的判定，同时调用教师选取作业示范模块的业务逻辑接口。教师选取作业示范模块业务逻辑层列表如表4-1.3所示。

表 4-2.3教师选取作业示范模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 选取示范作业 | HomeworkRecodService.java | 调用HomeworkRecodMapper | 返回给Api |

在教师选取作业示范模块的业务逻辑层是调用了公用的HomeworkRecodService接口，同时在实现该接口。

HomeworkRecodService.java接口主要方法：

HomeworkRecodService.java实现了HomeworkRecodService.java接口， 同时需要调用数据持久层的HomeworkRecodMapper.Java，实现对数据持久层接口的调用。

## 4.8、学生分数统计模块

学生分数统计模块的系统内部的相应响应操作示意图如图4-8所示。

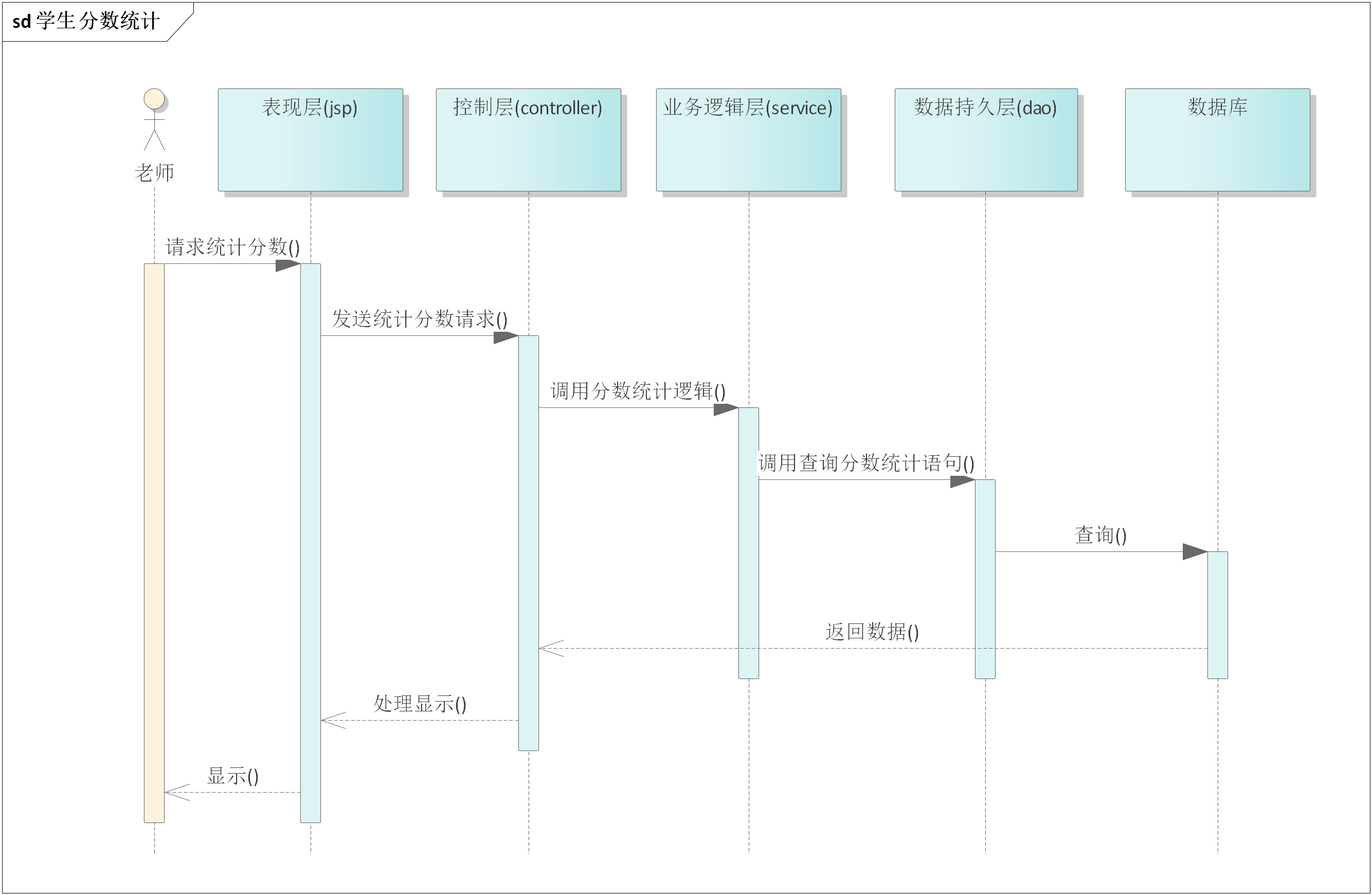


图 4-8 学生分数统计模块的信息响应示意图

### 4.8.1 表现层

学生分数统计模块的表现层主要完成对学生分数的统计功能，以表或多种统计图显示学生分数统计情况。表现层对应的jsp页面列表见表4-8.1所示。

表 4-8.1 学生分数统计表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | Jsp | 功能描述 |
| 学生分数统计界面 | statisticalScore.jsp | 用户的修改个人密码功能，当修改出错时给出提示。 |

学生分数统计模块的流程图如图4-8.1所示。



图 4-8 学生分数统计模块详细设计流程图

### 4.8.2 控制层

学生分数统计模块的控制层负责接受来自statisticalScore.jsp的教师分数统计请求，同时调用学生分数统计模块的业务逻辑接口，将学生分数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。学生分数统计模块控制层列表见表4-2.2所示。

表 4-2.2 学生分数统计模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 |  | 出口 |
| 学生分数统计 | HomeworkRecod.statisticalScore | 分数统计成功 | |

在控制层HomeworkRecod.statisticalScore的描述如下所示：

### 4.8.3 业务逻辑层

学生分数统计模块的业务逻辑层主要完成对学生分数统计逻辑的判定，同时调用学生分数统计模块的业务逻辑接口。学生分数统计模块业务逻辑层列表如图4-2.3所示。

表 4-8.3学生分数统计模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 学生分数统计 | HomeworkRecodService.java | 调用HomeworkRecodMapper | 返回给API |

在学生分数统计模块的业务逻辑层是调用了公用的HomeworkRecodService

接口，同时在实现该接口。

### 4.8.4 数据持久层

学生分数统计模块的是数据持久层能对已批改的作业数据进行增删改查。学生分数统计模块数据持久层列表见表4-8.4所示。

表 4-8.4 学生分数统计模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 学生分数统计 | HomeworkRecodMapper.java | HomeworkRecod.java | 对已批改作业信息进行增删改查操作 |

HomeworkRecod.java定义了学生作业信息进行增、删、改、查的接口

### 4.8.5 域模型层

学生分数统计模块用到域模型层中HomeworkRecod.java，HomeworkRecod.java是一个公用域模型，在涉及到学生作业查询等操作时，就会调用到该模型，学生分数统计模块域模型层列表如表4-8.5所示。

表 4-8.5学生分数统计模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| HomeworkRecod.java | 对学生作业信息 |

HomeworkRecod.java主要属性与方法：查看4-6.5

## 4.9、学生提交作业模块

学生提交作业模块的系统内部的相应响应操作示意图如图4-9所示。

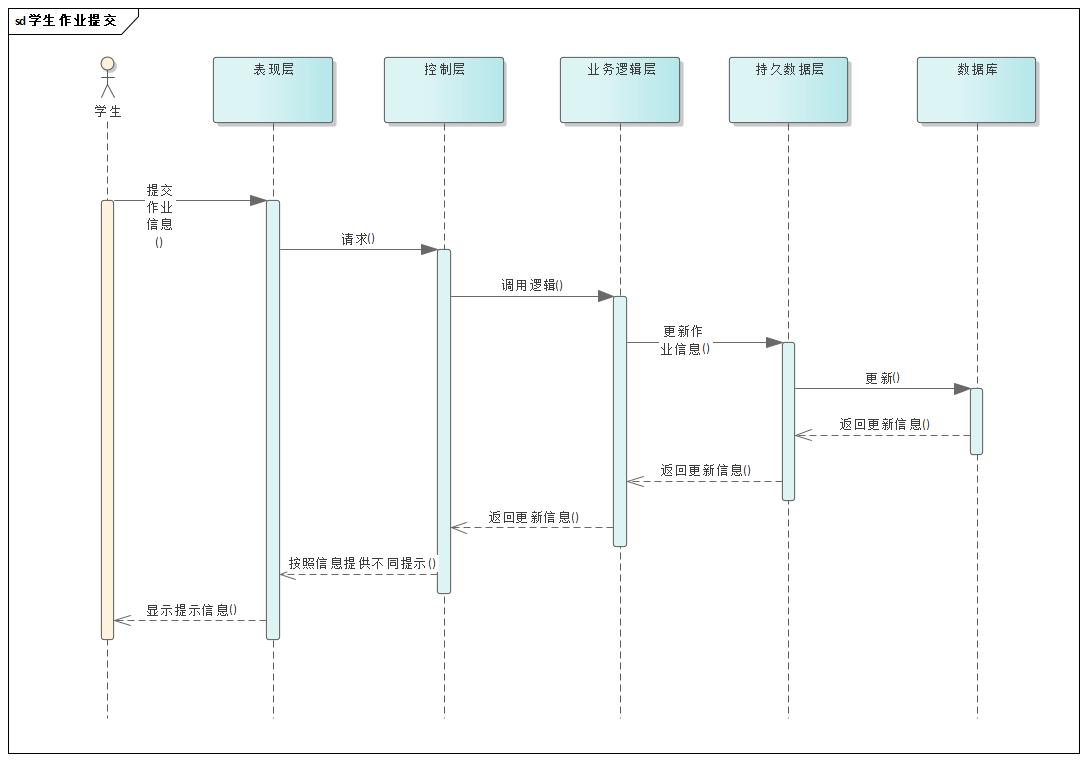


图 4-9 学生提交作业模块的信息响应示意图

### 4.9.1 表现层

学生提交作业模块的表现层主要完成学生提交作业的功能，确认提交后页面给出响应消息，提示提交成功或失败的提示。表现层对应的jsp页面列表见表4-9.1所示。

表 4-9.1 学生提交作业表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | Jsp | 功能描述 |
| 学生提交作业界面 | myHomework.jsp | 学生提交作业 |

学生提交作业模块的流程图如图4-9.1所示。



图 4-9 学生提交作业模块详细设计流程图

### 4.9.2 控制层

学生提交作业模块的控制层负责接受来自myHomework的作业，同时调用学生提交作业模块的业务逻辑接口，将作业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。学生提交作业模块控制层列表见表4-1.2所示。

表 4-1.2 学生提交作业模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 | 转移说明 | 出口 |
| 学生提交作业 | HomeworkInfo.doShortHomework | SUCCESS | myHomework提示提交成功 |
| ERROR | myHomework提示提交失败。 |

在控制层HomeworkInfo.doShortHomework功能描述如下所示

### 4.9.3 业务逻辑层

学生提交作业模块的业务逻辑层主要完成对学生提交作业逻辑的判定，同时调用学生提交作业模块的业务逻辑接口。学生提交作业模块业务逻辑层列表如表4-9.3所示。

表 4-9.3学生提交作业模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 学生提交作业 | HomeworkService.java | 调用HomeworkMapper | 返回给Api |

在学生提交作业模块的业务逻辑层是调用了公用的HomeworkService接口，同时在实现该接口。

HomeworkService.java接口主要方法：

HomeworkServiceImpl.java实现了HomeworkServiceImpl.java接口， 同时需要调用数据持久层的HomeworkMapper.Java，实现对数据持久层接口的调用。

### 4.9.4 数据持久层

### 4.9.5 域模型层

学生提交作业模块用到域模型层中SysUser.java，SysUser.java是一个公用域模型，当提交作业操作时，就会调用到该模型，学生提交作业模块域模型层列表如表4-9.5所示。

表 4-9.5登陆模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| HomeWorkRecord.java | 对学生作业信息的增、删、改、查操作 |

HomeWorkRecord.java主要属性与方法：查看查看4-6.5

## 4.10、学生查看批改情况模块

学生查看批改情况模块的系统内部的相应响应操作示意图如图4-10所示。

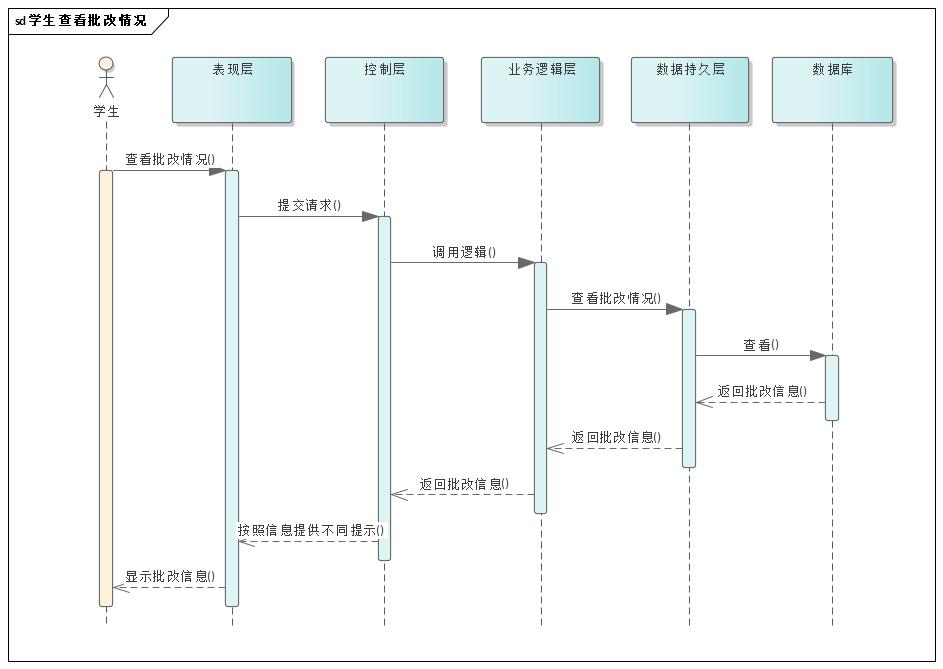


图 4-10 学生查看批改情况模块的信息响应示意图

### 4.10.1 表现层



图 4-10 学生查看批改情况模块详细设计流程图

### 4.10.2 控制层

学生查看批改情况模块的控制层负责调用学生查看批改情况模块的业务逻辑接口，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。学生查看批改情况模块控制层列表见表4-10.2所示。

表 4-10.2 学生查看批改情况

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 |  | 出口 |
| 学生查看批改情况 | Homework.results | homeworkResults.jsp | |

在控制层Homework.results功能描述如下所示

### 4.10.3 业务逻辑层

学生查看批改情况模块的业务逻辑层主要完成调用学生查看批改情况模块的业务逻辑接口。学生查看批改情况业务逻辑层列表如表4-10.3所示。

表 4-10.3学生查看批改情况模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 学生查看批改情况 | HomeworkService.java | 调用HomeworkMapper | 返回给Api |

在学生查看批改情况模块的业务逻辑层是调用了公用的IUserService接口，同时在实现该接口。

HomeworkServiceImpl.java实现了HomeworkService.java接口， 同时需要调用数据持久层的HomeworkMapper.Java，实现对数据持久层接口的调用。

### 4.10.4 数据持久层

学生查看批改情况模块的数据持久层能对批改的作业进行增删改查。学生查看批改情况模块数据持久层列表见表4-1.4所示。

表 4-1.4 学生查看批改情况模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 学生查看批改情况 | HomeworkMapper.java | 调用 HomeworkRecode.java | 对已批改作业进行查询操作 |

HomeworkMapper.java定义了批改作业查的接口：

### 4.10.5 域模型层

学生查看批改情况模块用到域模型层中HomeworkRecode.java，HomeworkRecode.java是一个公用域模型，在涉及到用户信息查询等操作时，就会调用到该模型，学生查看批改情况模块域模型层列表如表4-10.5所示。

表 4-10.5学生查看批改情况模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| HomeworkRecode.java | 对已批改作业信息查操作 |

HomeworkRecode.java主要属性与方法：查看4-6.5

## 4.11、学生查看作业示范模块

学生查看作业示范模块的系统内部的相应响应操作示意图如图4-10所示。

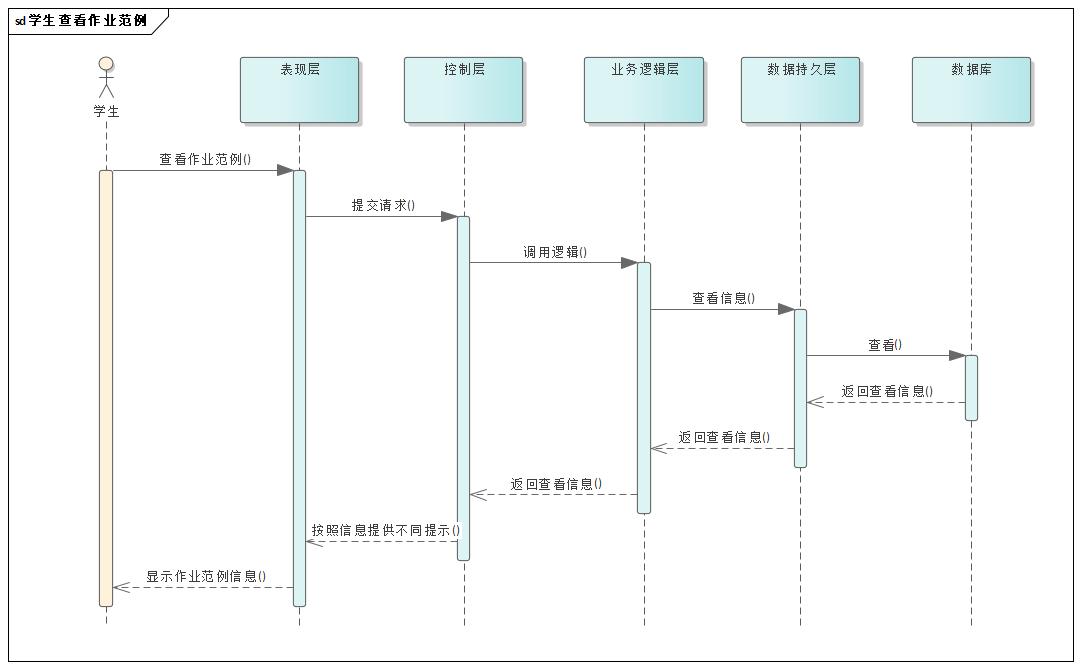


图 4-11 学生查看作业示范模块的信息响应示意图

### 4.11.1 表现层

学生查看作业示范模块的表现层主要完成对作业范例的查询功能，表现层对应的jsp页面列表见表4-6.1所示。

表 4-6.1 查询作业情况表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | jsp | 功能描述 |
| 学生查看作业示范界面 | DShomework.jsp | 查看作业范例的查询功能。 |

学生查看作业示范模块的流程图如图4-11.1所示。



图 4-11 学生查看作业示范模块详细设计流程图

### 4.11.2 控制层

学生查看作业示范模块的控制层负责接受来自DShomework.jsp的用户输入，同时调用学生查看作业示范模块的业务逻辑接口，将查询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。学生查看作业示范情况模块控制层列表见表4-11.2所示。

表 4-11.2 学生查看作业示范模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | 接口函数 | 转移说明 | 出口 |
| 学生查看作业示范 | homework. Demonstrate | SUCCESS | 查询成功，显示作业情况列表 |
| ERROR | 查询失败，提示对话框。 |

在控制层中homework. Demonstrate的描述如下所示：

### 4.11.3 业务逻辑层

学生查看作业示范模块的业务逻辑层主要完成对作业范例学习和点评逻辑的判定，同时调用作业范例学习和点评模块的业务逻辑接口。作业范例学习和点评模块业务逻辑层列表如图4-9.3所示。

表 4-9.3学生查看作业示范模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 查看作业范例 | HomeworkService.java | 调用HomeworkMapper | 返回给API |

在学生查看作业示范模块的业务逻辑层是调用了公用HomeworkService接口，同时在实现该接口。IExamRecordService.java接口主要方法：

HomeworkRecode getExcellentHomework();

### 4.11.4 数据持久层

学生查看作业示范模块的数据持久层是学生提交的作业表。学生查看作业示范模块数据持久层列表见表4-9.4所示。

表 4-9.4查看作业示范模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 查看作业示范模块 | HomeworkMapper.java | 调用 HomeWorkRecord.Java | 学生查看作业示范模块 |

HomeworkMapper.java定义了用户信息进行增、删、改、查的接口：

### 4.11.5 域模型层

学生查看作业示范模块用到域模型层中ExamRecord.java，在涉及到查看作业示范时，就会调用到该模型，学生查看作业示范模块域模型层列表如表4-11.5所示。

表 4-9.5作业范例学习和点评模块域模型层列表

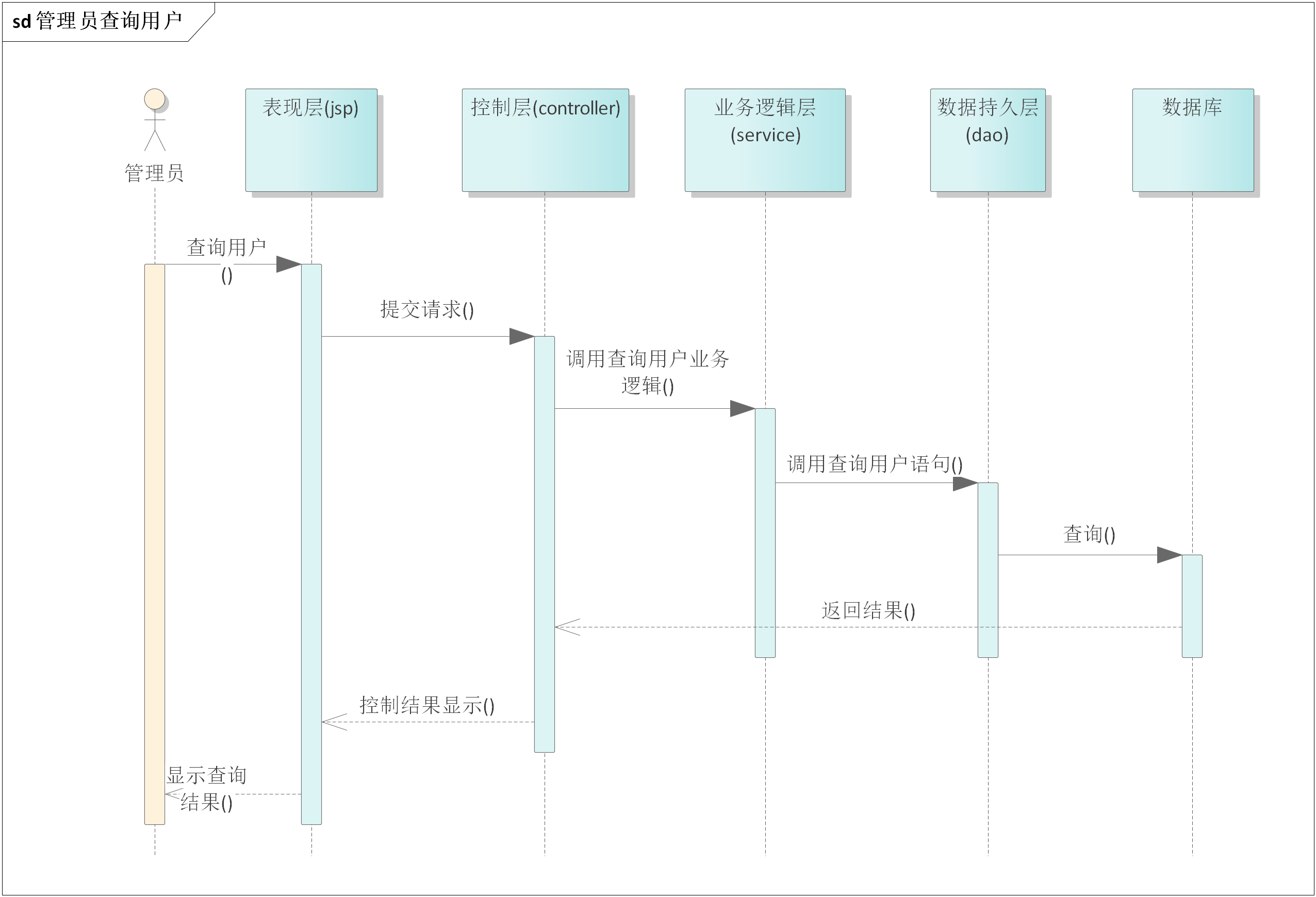
|  |  |
| --- | --- |
| 域模型 | 描述 |
| HomeWorkRecord.java | 学生查看作业示范 |

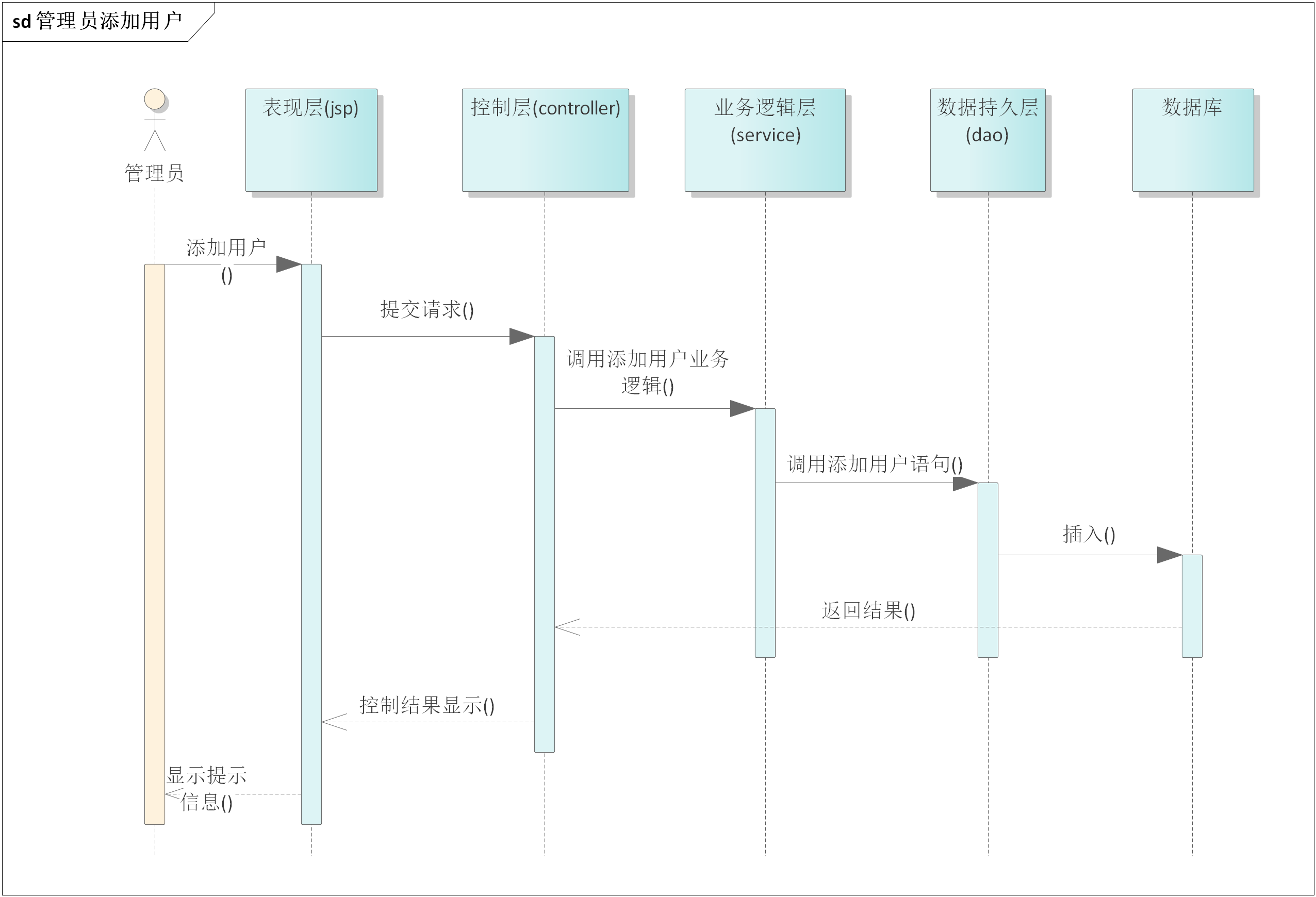
ExamRecord.java主要属性：查看4-6.5

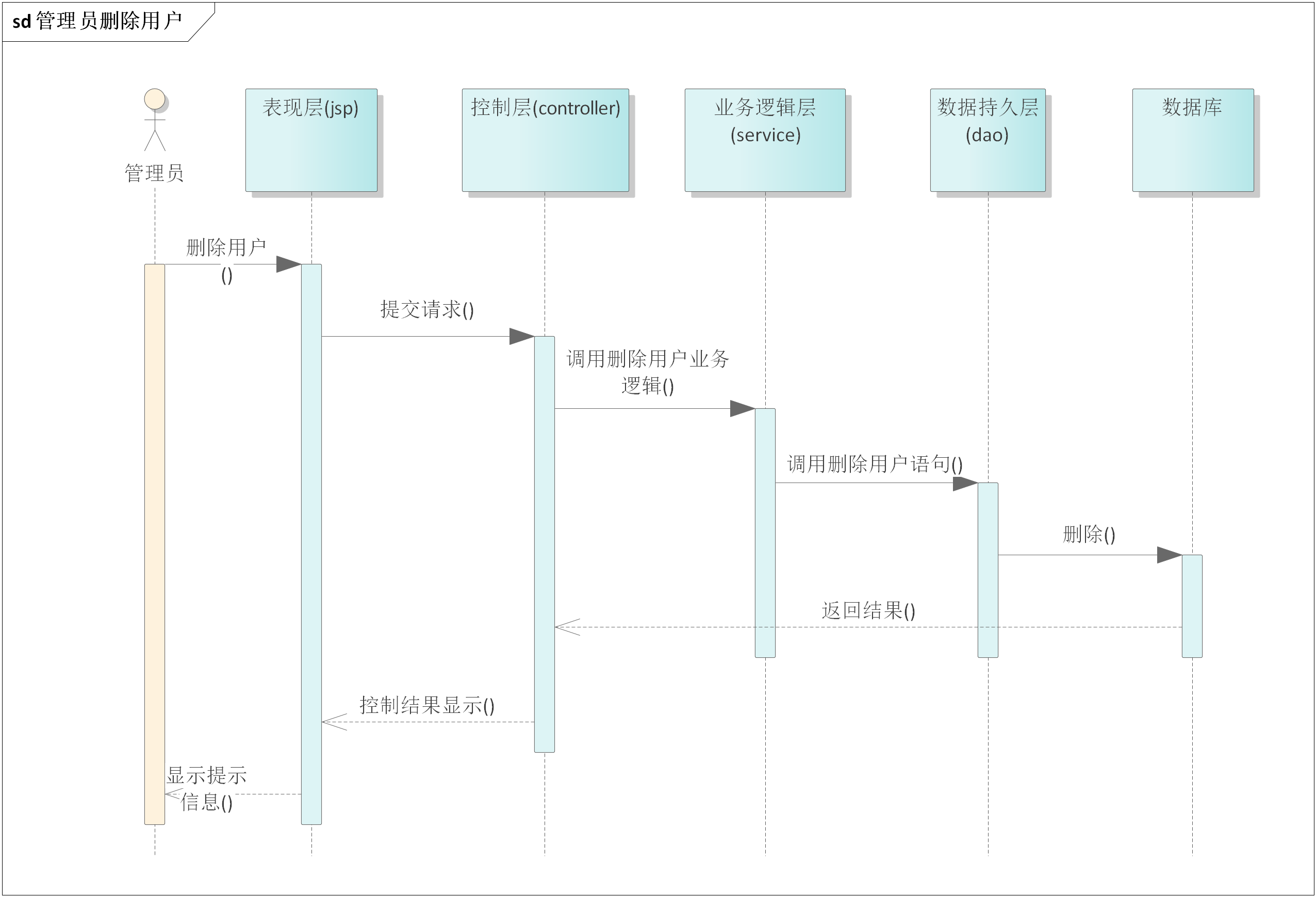
# 5、管理端模块详细设计

## 5.1、用户管理模块详细设计

用户管理模块的系统内部的相应响应操作示意图如图5-1所示。







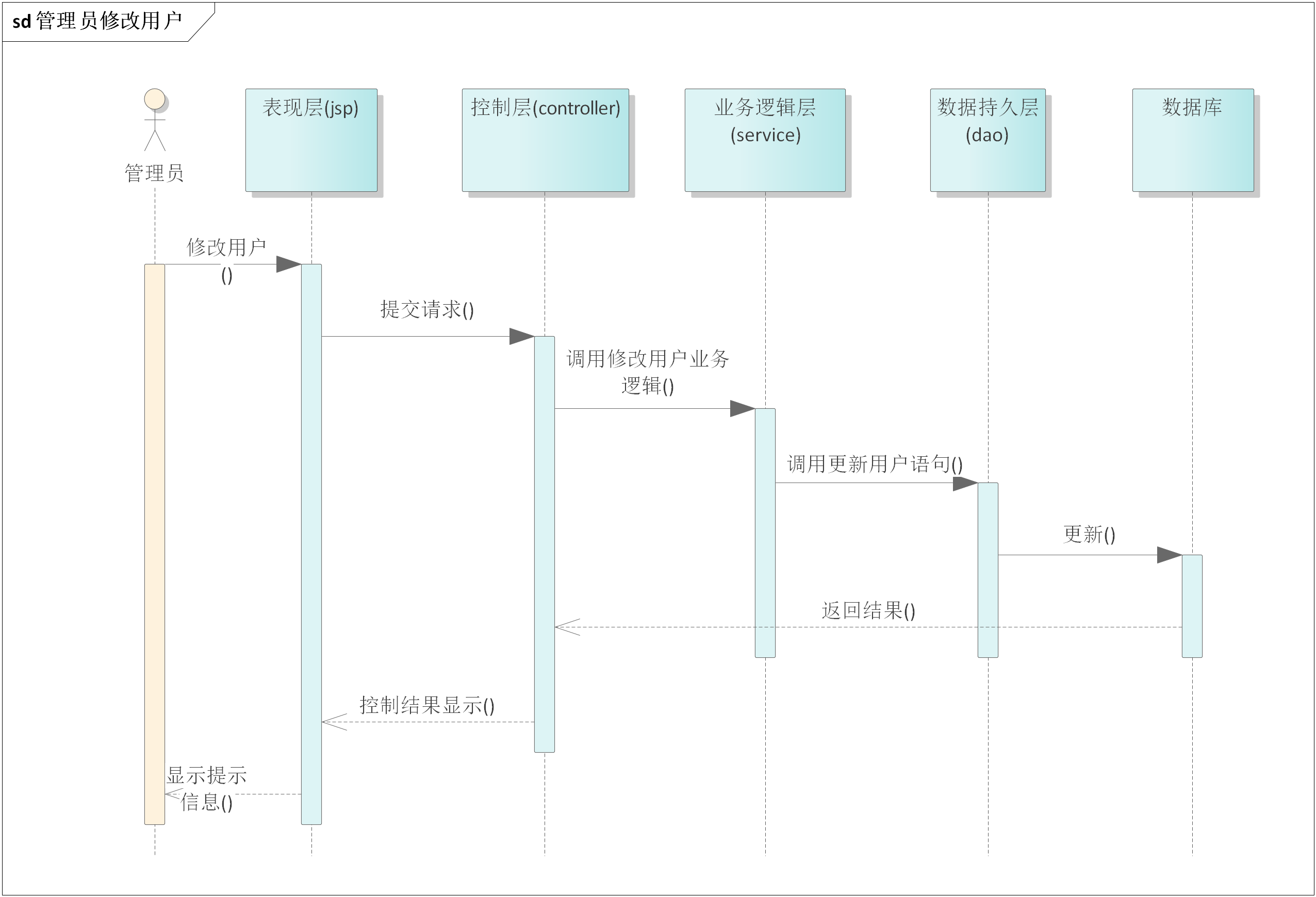


图5-3 用户管理模块响应操作示意图

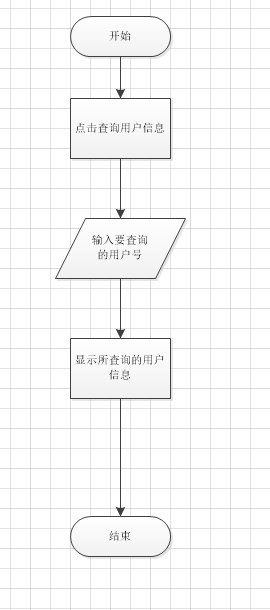
### 5.1.1 表现层

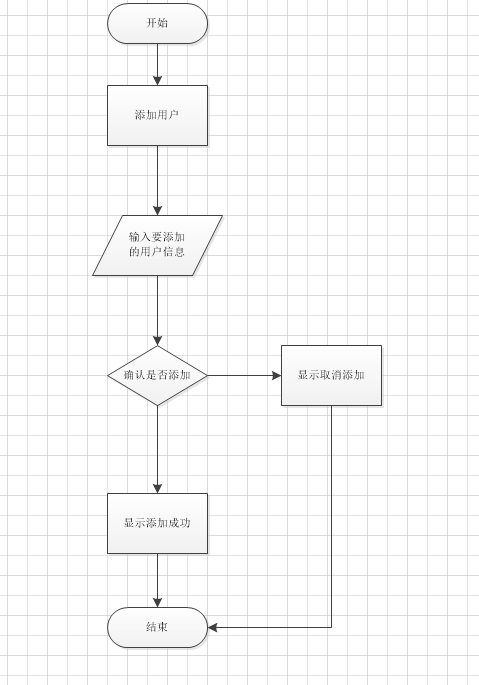
用户管理模块的表现层主要完成管理员对学生和教师的账号的增删查改功能，表现层对应的jsp页面列表见表5-1.1所示。

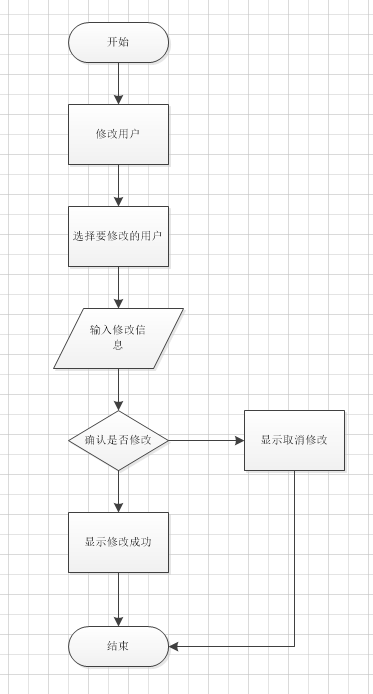
表 5-1.1 用户管理模块表现层jsp列表

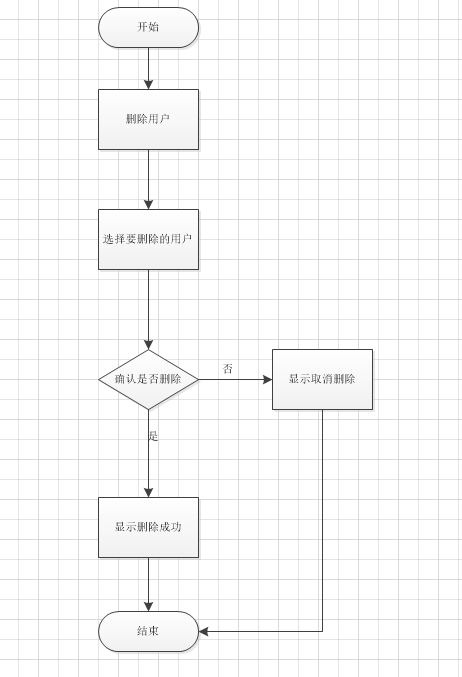
|  |  |  |
| --- | --- | --- |
| 界面 | Jsp | 功能描述 |
| 用户管理界面 | Student.jsp  Teacher.jsp | 完成管理员对学生和教师的账号的增删查改的管理功能。 |

用户管理模块的流程图如图5-3.1所示：









### 5.1.2 控制层

用户管理模块的控制层负责接受来自allStudent.jsp 、allTeacher.jsp的用户输入，同时调用用户管理的业务逻辑接口，将用户账号的增删查改操作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。账号管理模块控制层列表见表5-3.2所示。

表 5-1.2 用户管理模块控制层列表

|  |  |  |
| --- | --- | --- |
| 事件 | 接口 | 描述 |
| 用户管理 | StudentService.java  TeacherService.java | 学生、教师相关操作的API |

在控制层中StudentService.java、TeacherService.java的描述如下所示：

### 5.1.3 业务逻辑层

用户管理模块的业务逻辑层主要完成对用户管理逻辑的判定，同时调用用户管理模块的业务逻辑接口。用户管理模块业务逻辑层列表如图5-3.3所示。

表 5-3.3账号管理模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 用户管理 | TeacherService.java  StudentService.java | 调用StudentMapper、  TeacherMapper | 返回给API |

在用户管理模块的业务逻辑层是调用了公用的StudentService、TeacherService接口，同时在实现该接口。

StudentService.java、TeacherService.java接口主要方法：

public interface StudentMapper {

//添加一个学生

int addStudent(Student student);

//删除一个学生

int deleteStudentById(@Param("studentId") int id);

//修改学生信息

int updateStudent(Student student);

//查询一个学生

Student queryStudentById(@Param("studentId")int id);

//查询全部学生

List<Student> queryAllStudent();

//按照名字查询学生

List<Student> selectStudentByName(@Param("studentName")String StudentName);

}

public interface TeacherMapper {

//添加一个老师

int addTeacher(Teacher teacher);

//删除一个老师

int deleteTeacherById(@Param("teacherId") int id);

//修改老师信息

int updateTeacher(Teacher teacher);

//查询一个老师

Teacher queryTeacherById(@Param("teacherId")int id);

//查询全部老师

List<Teacher> queryAllTeacher();

//按名字查询老师

List<Teacher> selectTeacherByName(@Param("teacherName") String teacherName);}

5.1.4 数据持久层

用户管理模块的数据持久层是管理员对学生和教师信息的增删查改的管理。用户管理模块数据持久层列表见表5-1.4所示。

表 5-3.4用户管理模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 账号管理 | StudentMapper.java  TeacherMapper.java | 调用 Teacher.Java  Student.Java | 管理员对用户的增删查改的管理 |

### 5.1.5 域模型层

用户管理模块用到域模型层中student.java、taecher.java，student.java、taecher.java是公用域模型，在涉及到用户信息查询等操作时，就会调用到该模型，用户管理模块域模型层列表如表5-3.5所示。

表 5-3.5用户管理模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| Student.java、Taecher.java | 老师和学生的相关属性 |

public class Student {

private int studentId;

private String studentAccount;

private String studentName;

private String studentSex;

private String studentInstitute;

private String major;

private String studentClass;

private String studentPhone;

private String studentAuthenticationStatus;

private int studentAge;

}

public class Teacher {

private int teacherId;

private String teacherAccount;

private String teacherName;

private String teacherSex;

private String teacherPosition;

private String teacherPhone;

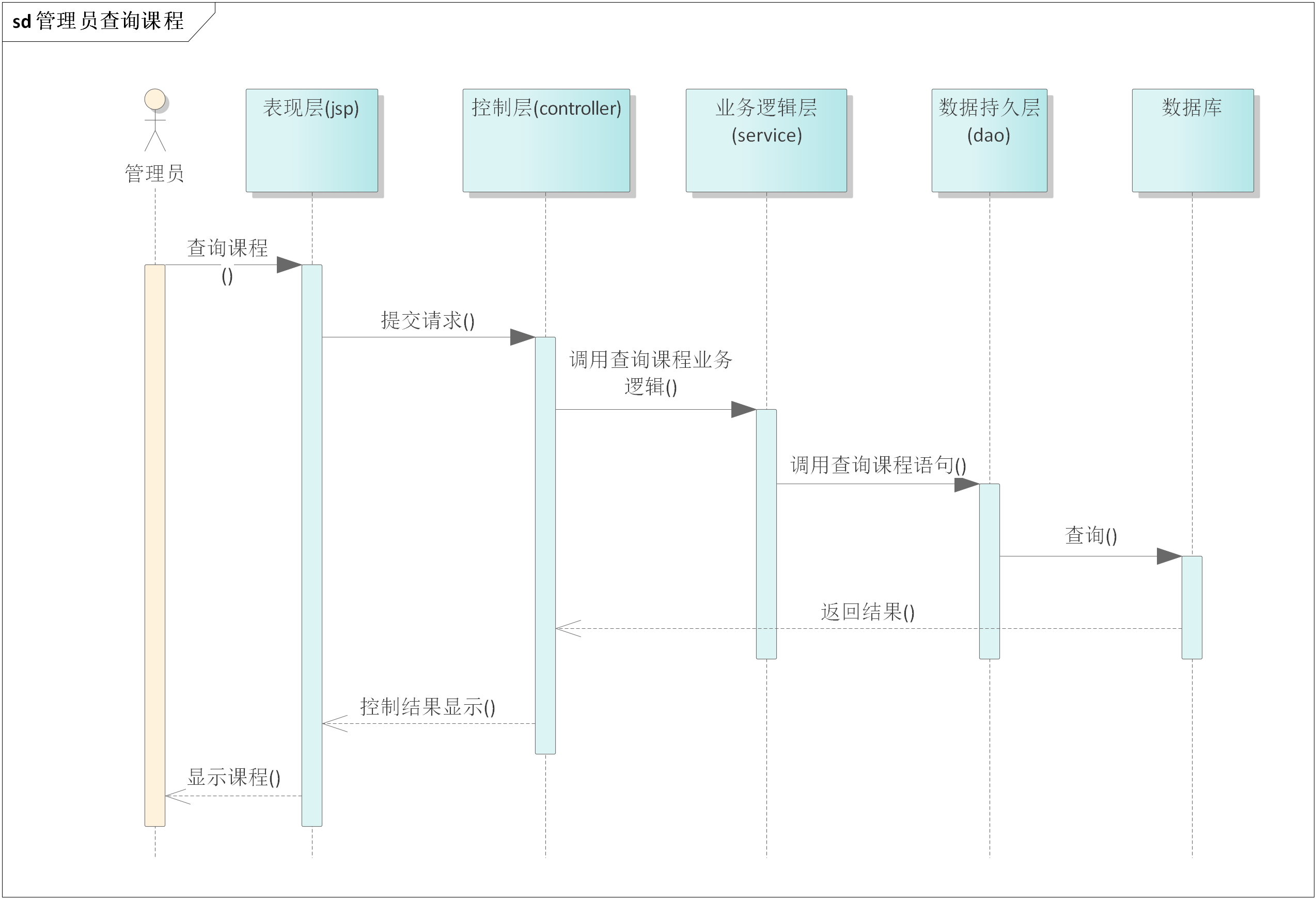
private String teacherEmail;

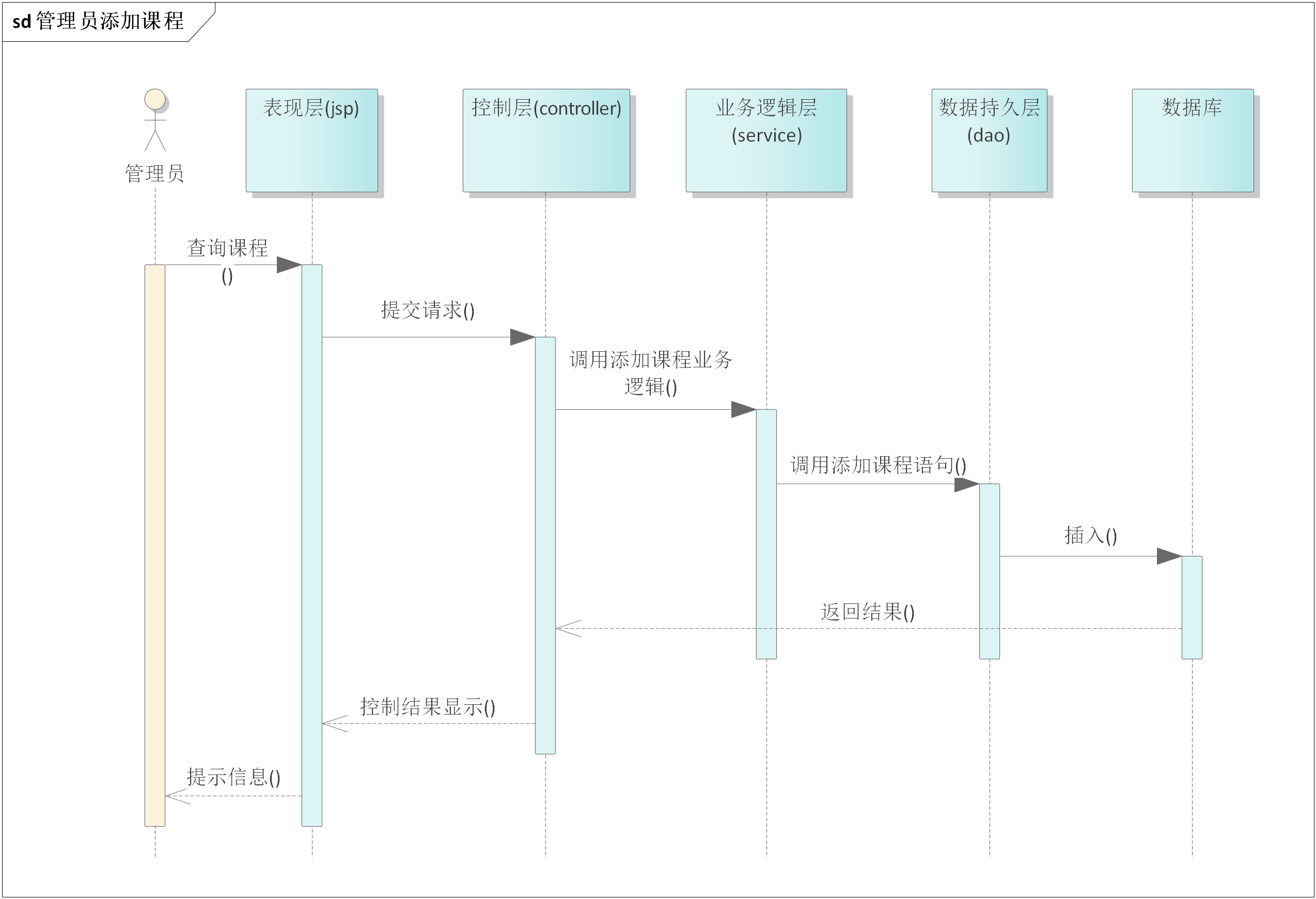
private int teacherAge;

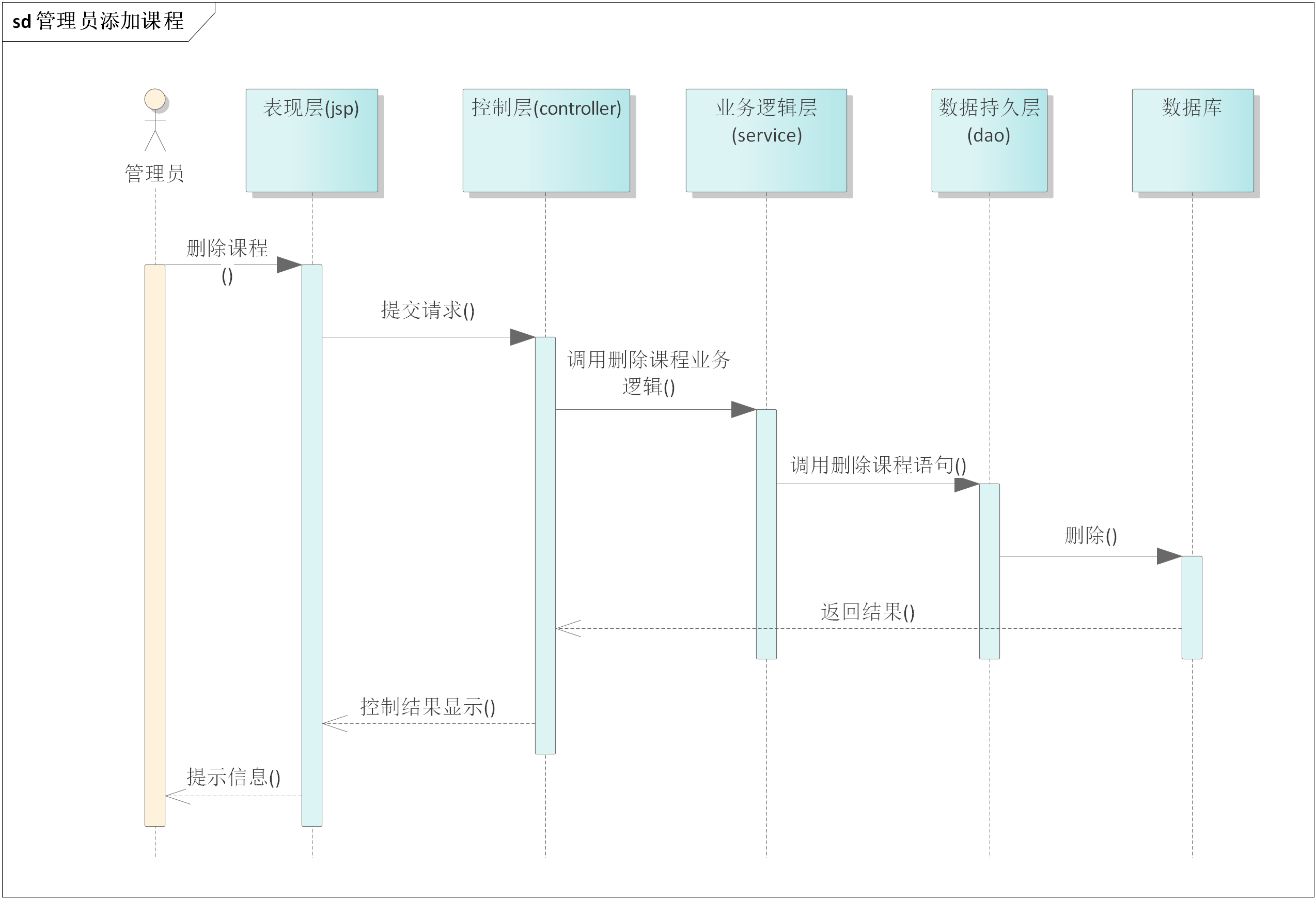
}

## 5.2、课程管理模块详细设计

课程管理模块的系统内部的相应响应操作示意图如图5-1所示。







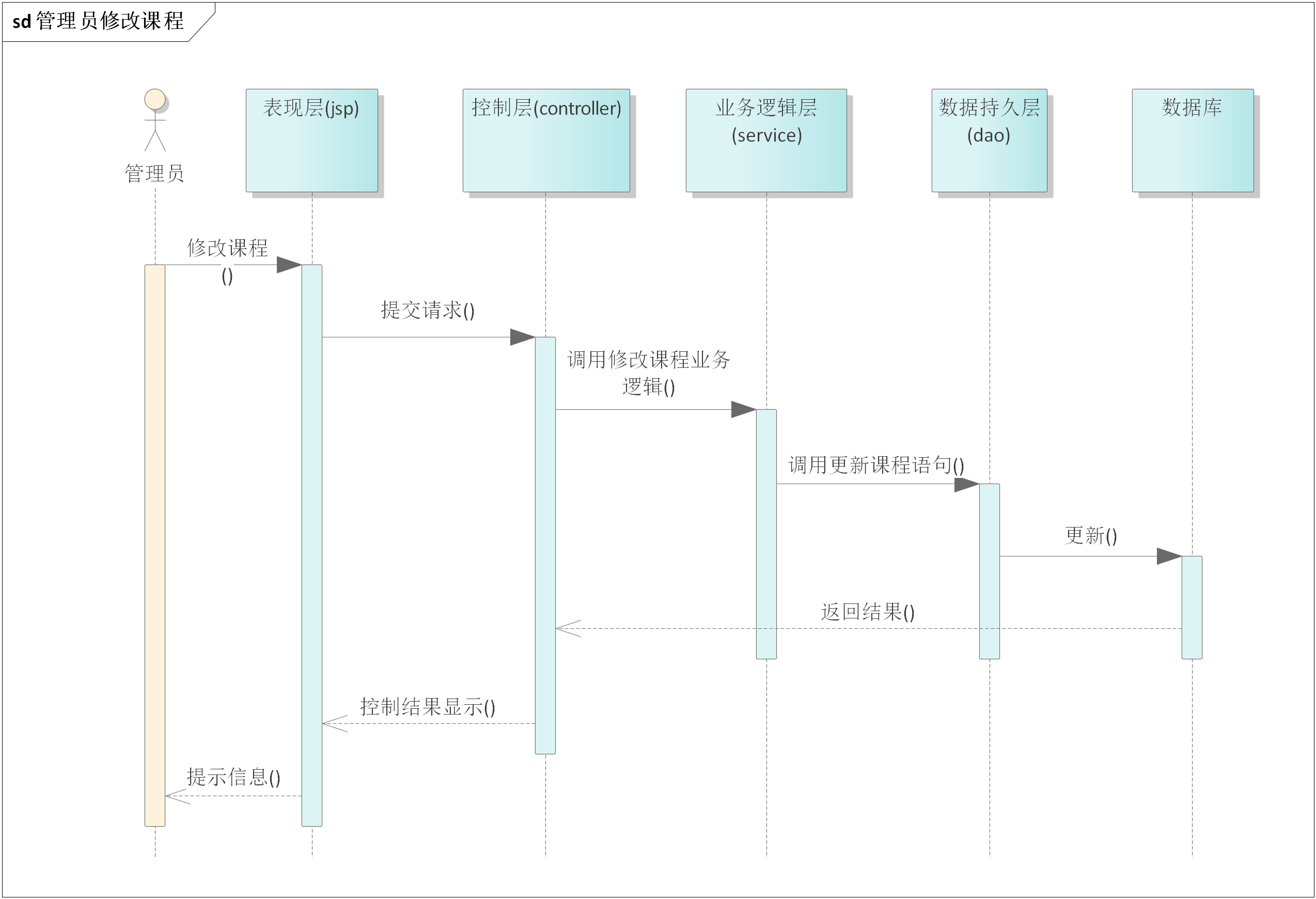


图5-2 课程管理模块响应操作示意图

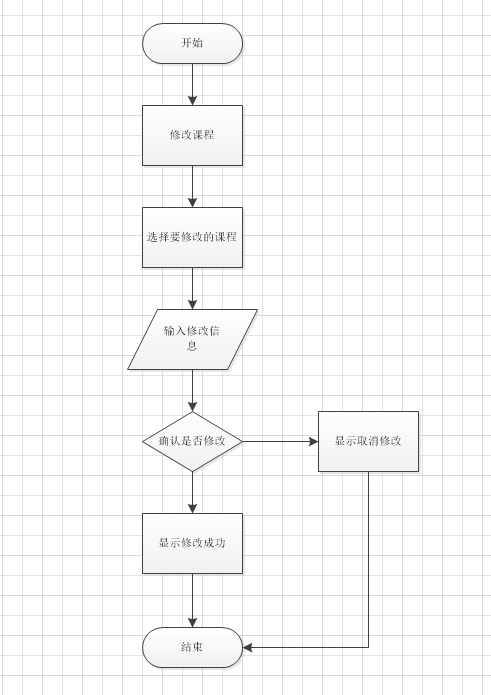
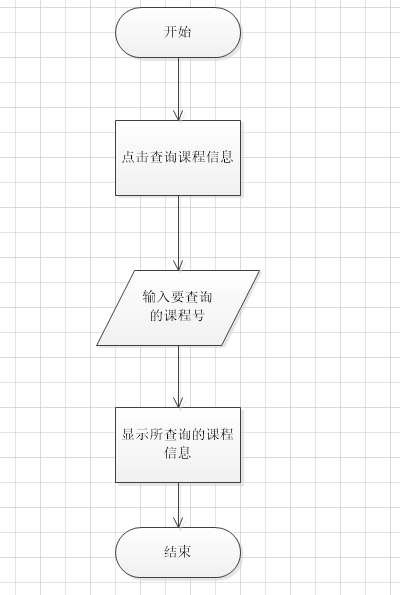
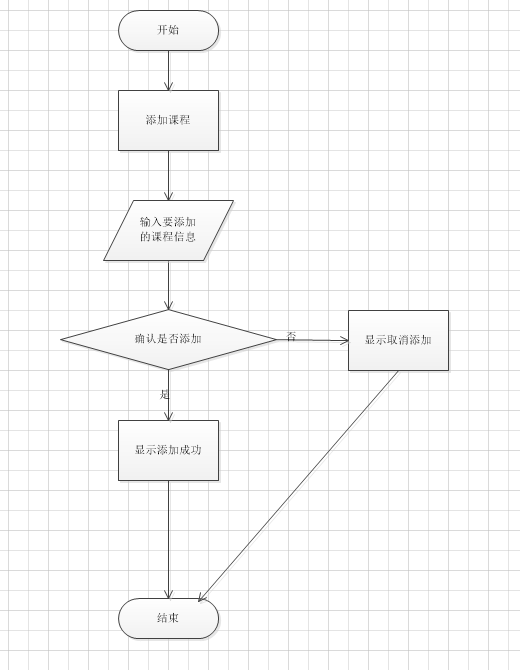
### 5.2.1 表现层

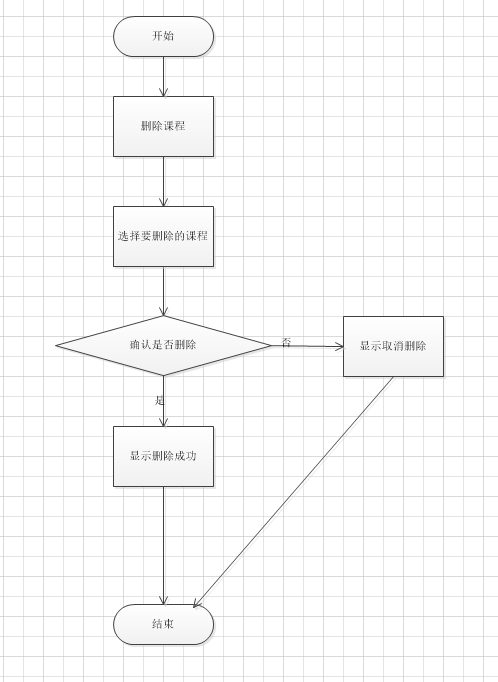
课程管理模块的表现层主要完成管理员对课程的增删查改功能，表现层对应的jsp页面列表见表5-2.1所示。

表 5-2.1 课程管理模块表现层jsp列表

|  |  |  |
| --- | --- | --- |
| 界面 | Jsp | 功能描述 |
| 课程管理界面 | coures.jsp | 完成管理员对课程信息的增删查改的管理功能。 |

课程管理模块的流程图如图5-2.1所示：





### 5.2.2 控制层

课程管理模块的控制层负责接受来自course.jsp 的用户输入，同时调用课程管理的业务逻辑接口，将课程的增删查改操作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。课程管理模块控制层列表见表5-3.2所示。

表 5-2.2 课程管理模块控制层列表

|  |  |  |
| --- | --- | --- |
| 事件 | 接口 | 描述 |
| 课程管理 | Course.selectAllCourse  Course.selectCourseByKey  Course.updateCourse  Course.deleteCourse  Course.addCourse | 管理员操作course |

在控制层中Course.selectCourse

Course.updateCourse

Course.deleteCourse

Course.addCourse的描述如下所示：

### 5.2.3 业务逻辑层

课程管理模块的业务逻辑层主要完成对课程管理逻辑的判定，同时调用课程管理模块的业务逻辑接口。课程管理模块业务逻辑层列表如图5-3.3所示。

表 5-2.3课程管理模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 课程管理 | CourserService.java | CourserMapper | 返回给控制层 |

在课程管理模块的业务逻辑层是调用了公用的CourserService、CourserServiceImpl接口，同时在实现该接口。

CourserServiceImpl接口主要方法：

List<Course> selectAllCourse()

Course selectCourseByKey(int courseId)

int updateCourse(Course course)

int deleteCourse(int courseId)

int addCourse(Course course)

### 5.2.4 数据持久层

课程管理模块的数据持久层是管理员对课程信息的增删查改的管理。课程管理模块数据持久层列表见表5-2.4所示。

表 5-2.4课程管理模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Mapper | 调用数据模型 | 说明 |
| 课程管理 | CourseMapper | 调用 Course.java | 管理员对课程的增删查改的管理 |

### 5.2.5 域模型层

课程管理模块用到域模型层中student.java、taecher.java，student.java、taecher.java是公用域模型，在涉及到课程信息查询等操作时，就会调用到该模型，课程管理模块域模型层列表如表5-2.5所示。

表 5-2.5课程管理模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| Course.java | 课程表实体类 |

public class Course {

private int courseId;

private String courseName;

private float credit;

}