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

**课程资料自主学习系统详细设计**

**V1.0**

**广西民族大学**

编写：邓振龙

日期： 2021年6月16日

目录

[1. 导言 5](#_Toc75514666)

[1.1目的 5](#_Toc75514667)

[1.2 范围 6](#_Toc75514668)

[1.3 引用标准 6](#_Toc75514669)

[1.4 参考资料 6](#_Toc75514670)

[1.5 版本更新信息 6](#_Toc75514671)

[2. 详细设计简述 7](#_Toc75514672)

[2.1 设计简介 7](#_Toc75514673)

[2.2模块简介 7](#_Toc75514674)

[3. 界面详细设计 9](#_Toc75514675)

[3.1 页面设计说明 9](#_Toc75514676)

[3.2 静态页面详细设计 10](#_Toc75514677)

[4. 客户端模块详细设计 10](#_Toc75514678)

[4.1 用户登录模块详细设计 10](#_Toc75514679)

[4.1.1 表现层 11](#_Toc75514680)

[4.1.2 控制层 13](#_Toc75514681)

[4.1.3 业务逻辑层 14](#_Toc75514682)

[4.1.4 数据持久层 16](#_Toc75514683)

[4.1.5 域模型层 18](#_Toc75514684)

[4.2 资料上传详细设计模块 19](#_Toc75514685)

[4.2.1 表现层 19](#_Toc75514686)

[4.2.2 控制层 20](#_Toc75514687)

[4.2.3 业务逻辑层 22](#_Toc75514688)

[4.2.4 数据持久层 23](#_Toc75514689)

[4.2.5 域模型层 24](#_Toc75514690)

[4.3 资料下载模块详细设计模块 25](#_Toc75514691)

[4.3.1 表现层 26](#_Toc75514692)

[4.3.2 控制层 27](#_Toc75514693)

[4.3.3 业务逻辑层 29](#_Toc75514694)

[4.3.4 数据持久层 30](#_Toc75514695)

[4.3.5 域模型层 31](#_Toc75514696)

[4.4密码修改详细设计模块 32](#_Toc75514697)

[4.4.1 表现层 33](#_Toc75514698)

[4.4.2 控制层 34](#_Toc75514699)

[4.4.3 业务逻辑层 36](#_Toc75514700)

[4.4.4 数据持久层 37](#_Toc75514701)

[4.4.5 域模型层 38](#_Toc75514702)

[5. 管理端模块详细设计 38](#_Toc75514703)

[5.1 系统用户管理详细设计模块 38](#_Toc75514704)

[5.1.1 表现层 39](#_Toc75514705)

[5.1.2 控制层 40](#_Toc75514706)

[5.1.3 业务逻辑层 42](#_Toc75514707)

[5.1.4 数据持久层 44](#_Toc75514708)

[5.1.5 域模型层 45](#_Toc75514709)

[5.2 课程信息管理模块详细设计 46](#_Toc75514710)

[5.2.1 表现层 47](#_Toc75514711)

[5.2.2 控制层 48](#_Toc75514712)

[5.2.3 业务逻辑层 49](#_Toc75514713)

[5.2.4 数据持久层 50](#_Toc75514714)

[5.2.5 域模型层 51](#_Toc75514715)

[5.3 网站新闻管理模块详细设计 52](#_Toc75514716)

[5.3.1 表现层 53](#_Toc75514717)

[5.3.2 控制层 54](#_Toc75514718)

[5.3.3 业务逻辑层 55](#_Toc75514719)

[5.3.4 数据持久层 56](#_Toc75514720)

[5.3.5 域模型层 56](#_Toc75514721)

[6. 公共部分模块详细设计 57](#_Toc75514722)

[6.1 查看网站新闻模块详细设计 57](#_Toc75514723)

[6.1.1 表现层 57](#_Toc75514724)

[6.1.2 控制层 59](#_Toc75514725)

[6.1.3 业务逻辑层 60](#_Toc75514726)

[6.1.4 数据持久层 61](#_Toc75514727)

[6.1.5 域模型层 62](#_Toc75514728)

[6.2 查看课程信息模块详细设计 62](#_Toc75514729)

[6.2.1 表现层 63](#_Toc75514730)

[6.2.2 控制层 64](#_Toc75514731)

[6.2.3 业务逻辑层 66](#_Toc75514732)

[6.2.4 数据持久层 67](#_Toc75514733)

[6.2.5 域模型层 68](#_Toc75514734)

# 导言

## 1.1目的

随着计算机技术的普及和计算机网络的普遍应用，学生自主学习，教师精心辅导也成趋势。建立一个网站方便更多的人享用宝贵的教育资料源，方便教师教学，提高学生自学能力，已成趋势。同时，也能促进老师与同学之间的交流，当遇到问题时，可以及时的联系到老师寻求帮助，大大提升了学习的效率。

该文档是对课程资料自主学习系统概要设计进行详细说明，是为了指导和规范课程资料自主学习系统开发而制定的详细开发设计文档，包括程序描述、输入/输出、算法和流程逻辑等，为软件编程和系统维护提供基础。

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

该文档内容涵盖课资料自主学习系统的所有功能模块的详细设计描述，是基于概要设计对所要开发的系统进行全方位的详细描述，为后续的开发做好准备。

## 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

[2] 《软件工程导论》

## 1.5 版本更新信息

本文档的更新记录如下表1-5所示。

表1-5 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 |
| 001 | 2021.6.16 | 0.1 | 全部 | 初始发布版本 |
| 002 | 2021.6.18 | 1.0 | 3章节 | 增加 |
| 003 | 2021.6.19 | 2.0 | 4章节 | 增加、修改 |
| 004 | 2021.6.21 | 3.0 | 4、5章节 | 增加 |
| 005 | 2021.6.22 | 4.0 | 4、5、6章节 | 增加、修改 |
| 006 | 2021.6.24 | 5.0 | 全部 | 增加、修改 |

# 2. 详细设计简述

## 2.1 设计简介

系统的结构图如图2-1所示。

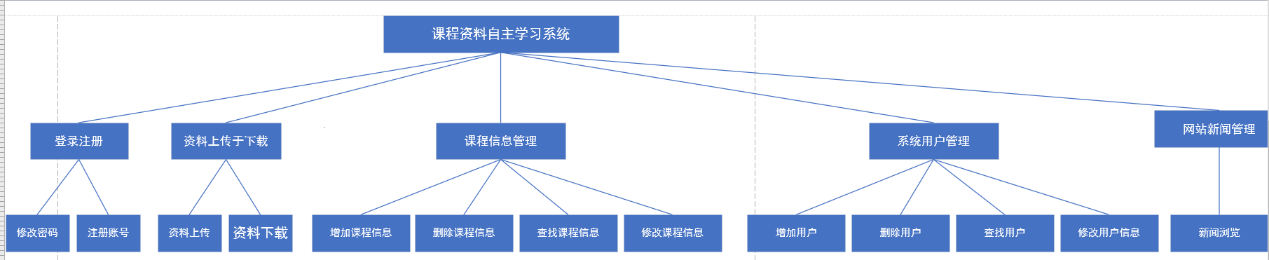


图2-1 系统的结构图

## 2.2 模块简介

系统的模块图如图2-2.1所示。

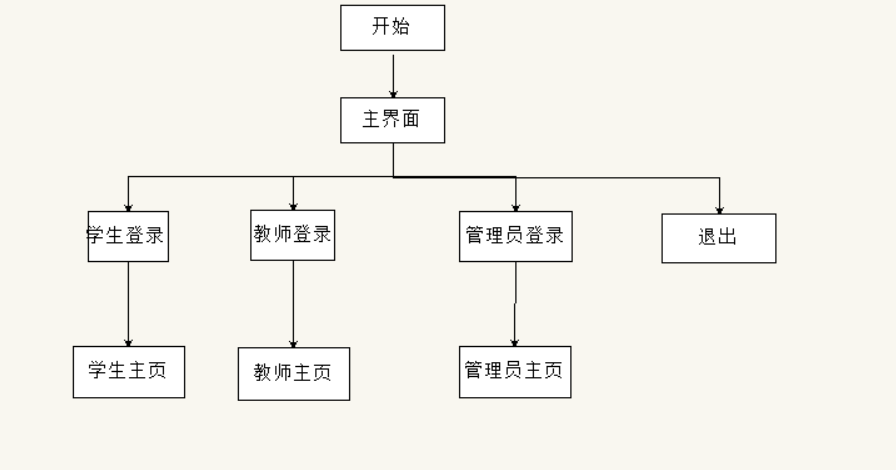


图2-2.1 系统的模块图

管理员主页的模块图如图2-2.2所示。

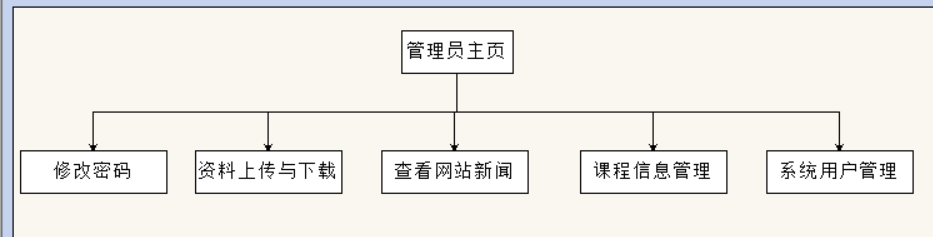


图2-2.2 管理员主页模块图

教师主页的模块图如图2-2.3所示。

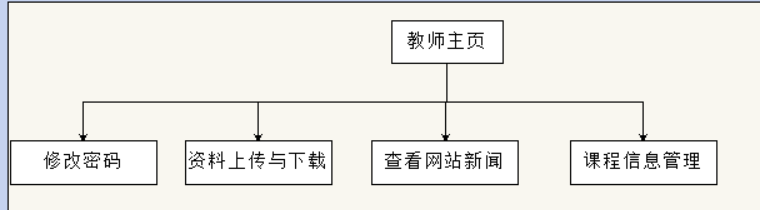


图2-2.3 教师主页模块图

学生主页的模块图如图2-2.4所示。

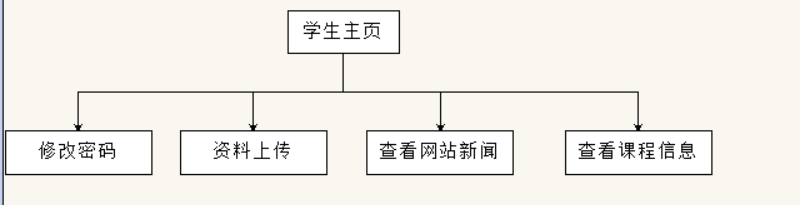


图2-2.4 学生主页模块图

# 3. 界面详细设计

## 3.1 页面设计说明

（1）保持一致性

在页面设计中要保持界面的一致性，包括使用标准的控件，使用相同的信息表现方式，如在图标字体标签风格颜色术语显示错误信息等方面的信息保持一致。统一的界面跳转规则和过渡动画，提示的方式等。

（2）布局合理性

网页在整体布局中要合理布局，把页面规划好，把可见的方框和线条都对齐，还要注意隐形的线条也要对齐。系统主题或者logo可重复出现，以提醒y用户现在正在使用此系统，达到强化系统主题和特色的目的。为了能更好的吸引用户，还要对合理的利用信息的区分，如新闻的标题和内容的字体的大小粗细，颜色的对比，疏密的对比，如在所要展示的页面可以提供一个颜色和周围元素及背景颜色差异明显的下载按钮，以此吸引更多的注意。

（3）其他

页面设计采用jsp css javascprit javaMVC模式技术，使用DIV+CSS式的排版方式。利用Apache的开源工具common-fileupload，实现资料的上传与下载。对文字的编排，在网页中会出现大量的文字，文字是网页必不可缺少的元素。对网页色彩，网页内容要合理的使用和排版。

## 3.2 静态页面详细设计

课程资料自主学习系统静态页面表如3-2所示：

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 登录页面 | login.jsp | 进行登录的页面 |
| 主页 | index | 登录系统后显示的页面 |

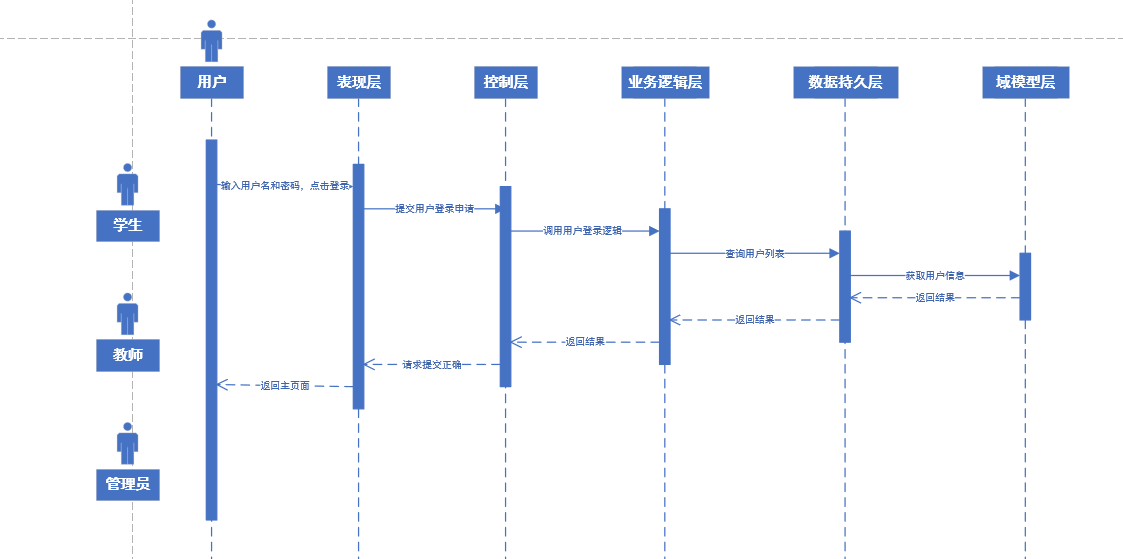
表3-2 页面表现层设计

# 4. 客户端模块详细设计

客户端模块主要包括用户登录模块、资料上传模块、资料下载模块、密码修改模块

## 4.1 用户登录模块详细设计

在登录模块时，系统内部的相应响应操作流程图如图4-1所示。



操作流程图4-1

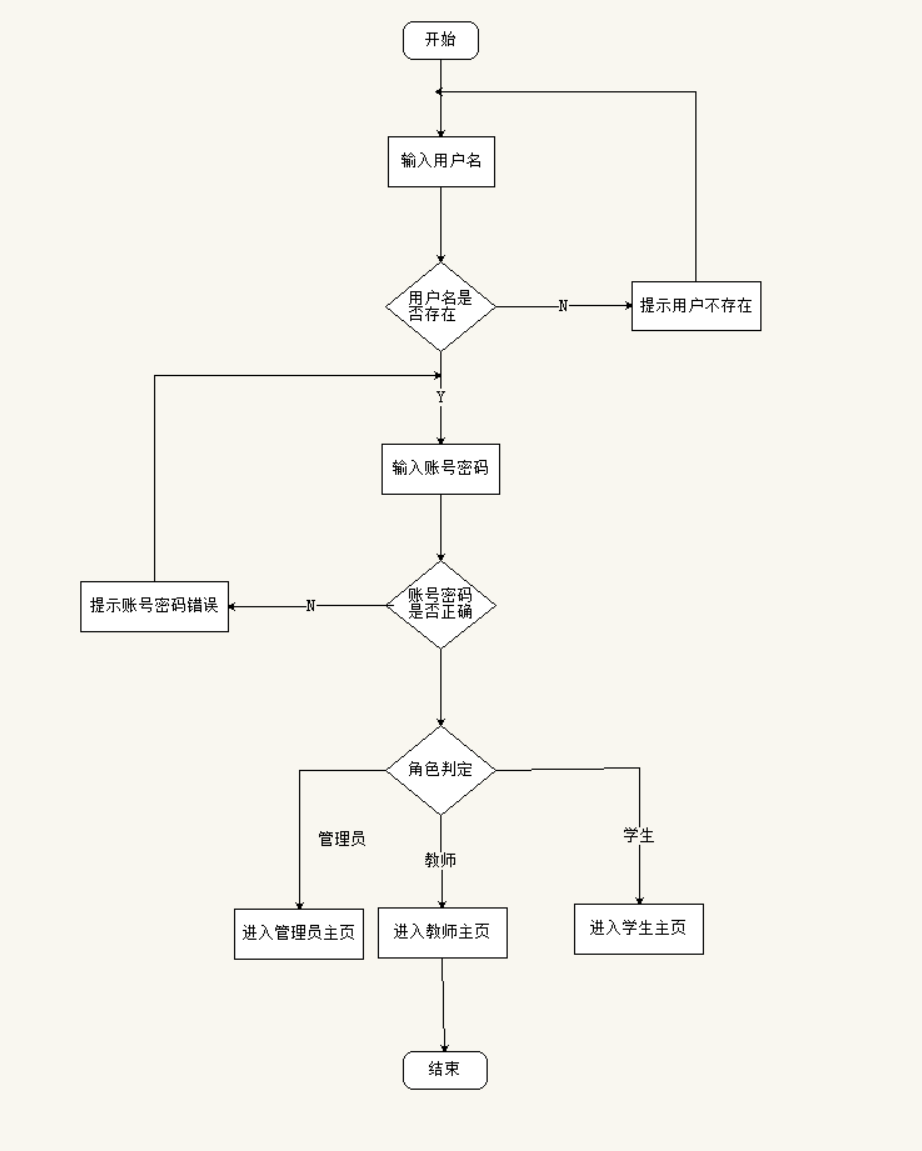
### 4.1.1表现层

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4-1-1.1所示

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

表 4-1-1.1 登入模块表现层JSP列表

login.jsp的流程图如图4-1-1.2所示



登录注册流程图4-1-1.2

### 4.1.2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4-1-2所示

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

**package** com.buptsse.spm.action;

/\*\*

\* *@description* 实现登录页面Action

\*/

**public** **class** LoginAction **extends** ActionSupport

{ **private** User user;

*@Resource*

**private** IUserService userService;

/\*实现各种属性的setter和getter方法\*/

/\*实现登入模块的控制层方法\*/

**public** String login(){}

}

### 4.1.3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4-1-3所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 登录 | IUserService.java  UserServiceImpl.java | 调用IUserDao | 返回给Action |

在登入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。IUserService.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** IUserService {

/\*登入检测\*/

**public** User login(String userName,String password);

/\*查找用户\*/

**public** User findUser(String userName);

/\*删除用户\*/

**public** boolean deleteUser(String username);

/\*增加用户\*/

**public** boolean addUser(User user);

}

UserServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** UserServiceImpl **implements** IUserService {

*@Resource*

**private** IUserDao iUserDao;

*@Override*

**public** User login(String userName,String password);

*@Override*

**public** User findUser(String userName);

*@Override*

**public** boolean deleteUser(String username);

*@Override*

**public** boolean addUser(User user);

}

### 4.1.4数据持久层

登录模块的是数据持久层能对用户数据进行增删改查。登录模块数据持久层列表见表4-1-4所示。

表 4-1-4 登录模块数据持久层列表

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

IUserDao.java定义了用户信息进行增、删、改、查的接口：

package com.buptsse.spm.dao;

/\*\*

\* @description 用户信息持久层接口定义，包括用户增加，保存，查询

\*/

**public** **interface** IUserDao {

**public** User findUser(User user); /\*查找用户信息\*/

**public** boolean insertUser(User user); /\*插入用户信息\*/

**public** List<User> searchUser(List<Object> choose); /\*查找用户信息\*/

**public** boolean deleteUser(String username); /\*删除用户信息\*/

**public** boolean addUser(User user); /\*增加用户信息\*/

}

### 4.1.5域模型层

登录模块用到域模型层中User.java，User.java是一个公用域模型,登录模块域模型层列表如表4-1-5所示。

|  |  |
| --- | --- |
| 域模型 | 描述 |
| User.java | 对用户信息的增、删、改、查操作 |

User.java主要属性与方法：

**package** com.buptsse.spm.domain;

/\*\*

\* *@description* User表的信息记录

\*/

**public** **class** User **implements** Serializable{

**private** **String** id; /\*id识别\*/

**private** **String** userName; /\*用户名\*/

**private** **String** password; /\*密码 \*/

**private** **String** password1; /\*验证密码\*/

**private** **String** position; /\*身份信息 \*/

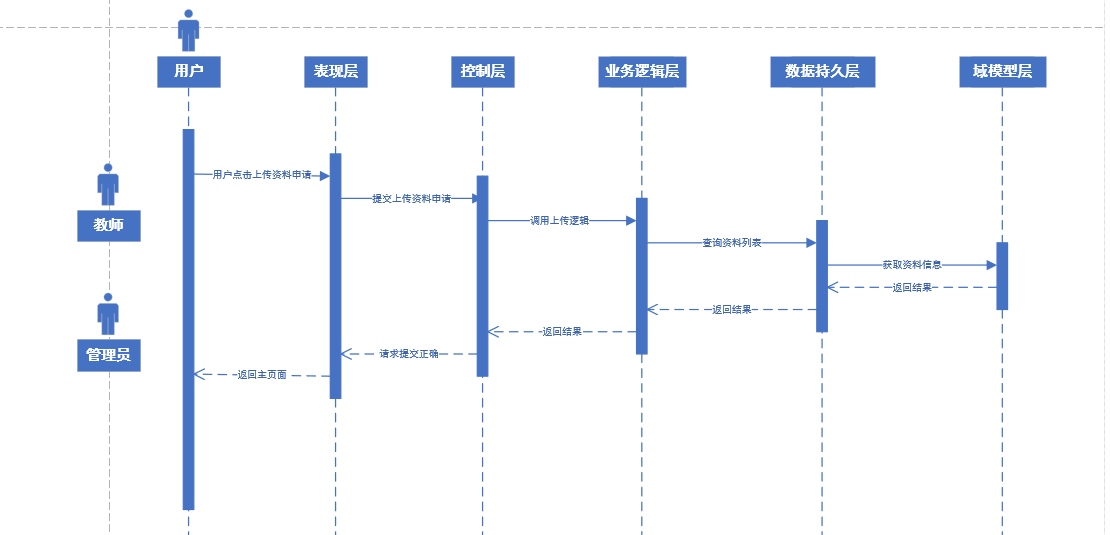
**private** **String** userId; /\*用户ID\*/

/\*编写对上述各个属性的*getter*与*setter*方法\*/

}

## 4.2 资料上传详细设计模块

在资料上传时，系统内部的相应响应操作示意图如图4-2所示。



操作流程图4-2

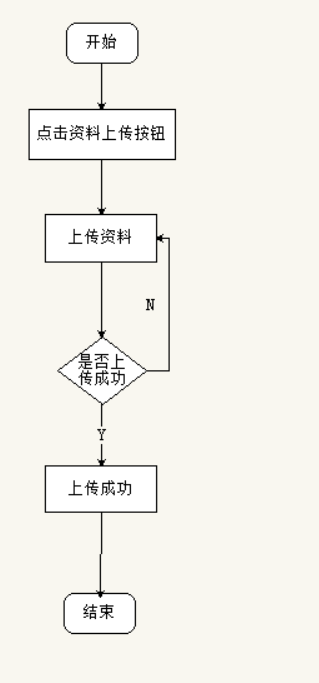
### 4.2.1表现层

资料上传模块的表现层主要是实现用户对资料的上传功能。

表现层对应的JSP页面列表见表4-2-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 资料上传页面 | upload.jsp | 用户进行资料上传 |

流程图如图4-2-1所示。



流程图4-2-1

### 4.2.2控制层

资料模块的控制层负责接受来自upload.jsp的用户上传操作，同时调用资料上传模块的业务逻辑接口，将文件名、文件大小、文件格式等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4-2-2所示。

表 4-2-2 资料上传模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 上传 | UploadAction.java | SUCCESS | upload.jsp—上传成功，显示提示窗口。 |
| ERROR | upload.jsp---上传失败，显示提示窗口。 |

在控制层中UploadAction.java的描述如下所示：

**package** com.buptsse.spm.action;

/\*\*

\* *@description* 实现上传页面Action

\*/

**public** **class**UploadAction **extends** ActionSupport

{

**private**Upload upload;

*@Resource*

**private** UploadService uploadService;

/\*实现各种属性的setter和getter方法\*/

/\*实现登入模块的控制层方法\*/

**public** String upload (){}

}

### 4.2.3业务逻辑层

资料上传模块的业务逻辑层主要完成对用户上传逻辑的判定，同时调用资料上传模块的业务逻辑接口。资料上传模块业务逻辑层列表如图4-2-2所示。

表 4-2-3资料上传模块业务逻辑层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 上传 | UploadService.java  UploadServiceImpl.java | 调用UploadDao | 返回给Action |

在资料上传模块的业务逻辑层是调用了公用的UploadService接口，同时在实现该接口。

UploadService.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** UploadService {

/\*上传检测\*/

**public** User upload(String fileName,String fileSize,String fileFormat);

/\*上传资料\*/

**public** String uploadFile(User user);

}

UploadServiceImpl.java实现了UploadService.java接口， 同时需要调用数据持久层的UploadDao.Java，利用@Resource来实现对数据持久层接口的调用。

UploadServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** UploadServiceImpl **implements** UploadService {

*@Resource*

**private** UploadDao UploadDao;

*@Override*

**public** Upload upload(String fileName,String fileSize,String fileFormat);

*@Override*

**public** String uploadFile(User user);

}

### 4.2.4数据持久层

资料上传模块的是数据持久层能对资料进行上传。资料上传模块数据持久层列表见表4-2-4所示。

表 4-2-4 资料上传模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 上传 | UploadDao.java  UploadDaoImpl.java | 调用 upload.Java | 用户进行上传操作 |

UploadDao.java

package com.buptsse.spm.dao;

**public** **interface** UploadDao {

**public** Uplaod uploadFile(String fileName，String format); /\*上传资料\*/

}

### 4.2.5域模型层

资料上传模块用到域模型层中Upload.java，Upload.java是一个公用域模型，登录模块域模型层列表如表4-2-5所示。

表 4-2-5资料上传模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| Upload.java | 用户进行上传操作 |

Upload.java主要属性与方法：

**package** com.buptsse.spm.domain;

/\*\*

\* *@description* Uplaod表的信息记录

\*/

**public** **class** Uplaod **implements** Serializable{

**private** **String** fileName; /\*文件名\*/

**private** **String** fileSize; /\*文件大小 \*/

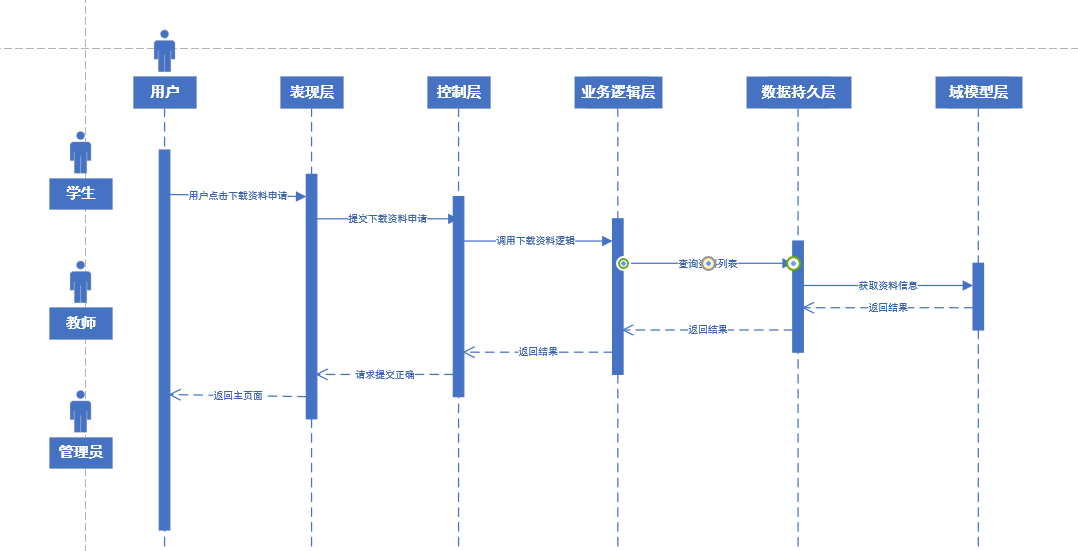
**private** **String** fileFormat; /\*文件格式\*/

/\*编写对上述各个属性的*getter*与*setter*方法\*/

}

## 4.3 资料下载模块详细设计模块

在资料下载模块时，系统内部的相应响应操作流程图如图4-3所示。



操作流程图4-3

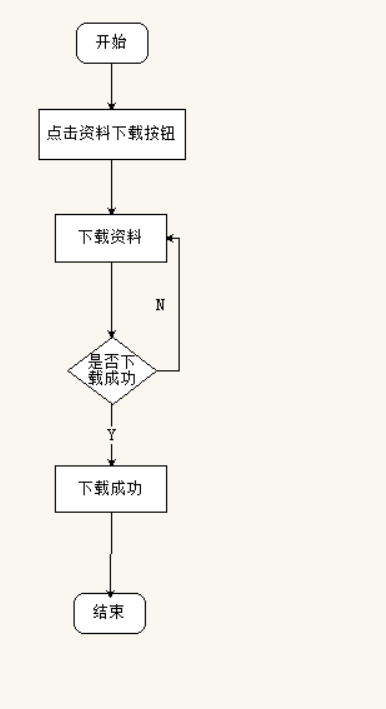
### 4.3.1表现层

资料下载模块的表现层主要是实现用户对资料的下载功能。

表现层对应的JSP页面列表见表4-3-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 资料下载页面 | Download.Jsp | 用户进行资料下载 |

流程图如图4-3-1所示。



流程图4-3-1

### 4.3.2控制层

资料下载模块的控制层负责接受来自用户的文件下载，同时调用资料下载业务逻辑接口，将文件的名字、格式、大小等关键信息传递到业务逻辑层进行判定。等到业务逻辑层处理完毕后，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。资料下载模块控制层列表见表4-3-2所示。

表 4-3-2 资料下载模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 下载 | DownloadAction.java | SUCCESS | mainFrame.jsp—下载成功，显示提示窗口。 |
| ERROR | download.jsp---下载失败，显示提示窗口。 |

在控制层中DownloadAction.java的描述如下所示：

**package** com.buptsse.spm.action;

/\*\*

\* *@description* 实现资料下载页面Action

\*/

**public** **class** Download Action **extends** ActionSupport

{

**private** Download download;

*@Resource*

**private** Download Service download Service;

/\*实现各种属性的setter和getter方法\*/

/\*实现资料上传模块的控制层方法\*/

**public** String download (){}

}

### 4.3.3业务逻辑层

资料下载模块的业务逻辑层主要完成对用户资料下载逻辑的判定，同时调用资料下载模块的业务逻辑接口。比如：用户下载的文件是否存在、格式是否正确，大小是否符合。资料下载模块业务逻辑层列表如图4-3-3所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 下载 | Download Service.java  DownloadServiceImpl.java | 调用Download Dao | 返回给Action |

DownloadService.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** download Service {

/\*上传检测\*/

**public** Download download (String fileName,String format,String size);

/\* 下载资料\*/

**Public** Download downloadFile(String String fileName);

}

DownloadServiceImpl.java实现了IUserService.java接口， 同时需要调用数据持久层的DownloadDao.Java，利用@Resource来实现对数据持久层接口的调用。

DownloadServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** Download ServiceImpl **implements** Download Service {

*@Resource*

**private** DownloadDao dsownload Dao;

*@Override*

**public** Download download(String fileName,String format,String size);

*@Override*

**public** Download downloadFile(String String fileName);}

### 4.3.4数据持久层

资料下载模块的是数据持久层能对资料的下载。资料下载模块数据持久层列表见表4-3-4所示。

表 4-3-4 资料下载模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 下载 | Download Dao.java  Download DaoImpl.java | 调用 download.Java | 对资料进行下载 |

DownloadDao.java

package com.buptsse.spm.dao;

**public** **interface** DownloadrDao {

**public** Download downloadFile(Download download); /\*资料下载\*/

}

### 4.3.5域模型层

资料上传模块用到域模型层中Download.java，资料上传模块域模型层列表如表4-3-5所示。

表 4-3-5登入模块域模型层列表

|  |  |
| --- | --- |
| 域模型 | 描述 |
| Download.java | 对资料进行下载操作 |

Download.java主要属性与方法：

**package** com.buptsse.spm.domain;

**public** **class** Download **implements** Serializable{

**private** **String** fileName; /\*文件名\*/

**private** **String** size; /\*文件大小\*/

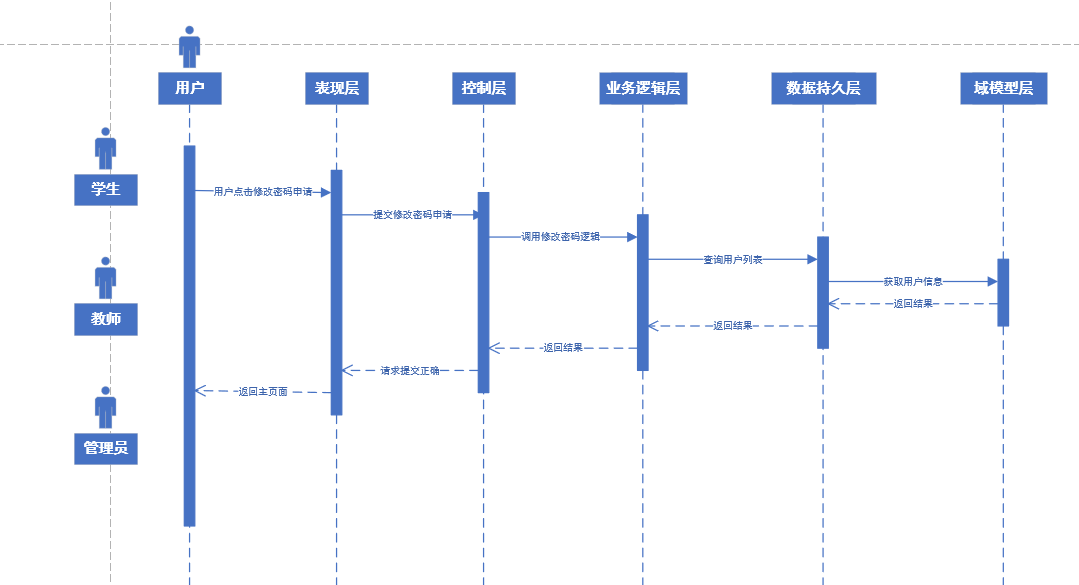
**private** **String** format; /\*文件格式\*/

/\*编写对上述各个属性的*getter*与*setter*方法\*/

}

## 4.4密码修改详细设计模块

在密码修改模块时，系统内部的相应响应操作流程图如图4-3所示。



操作流程图4-3

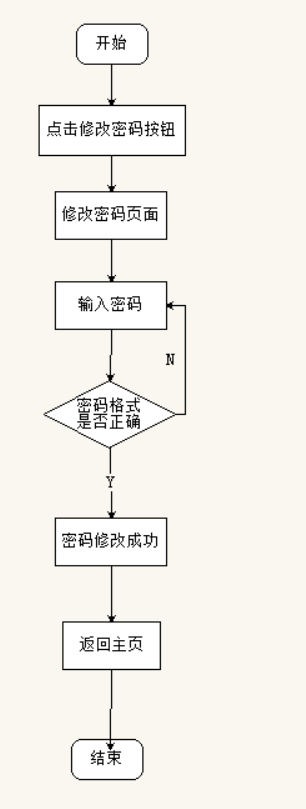
### 4.4.1表现层

密码修改模块的表现层主要是实现用户对密码的修改功能。

表现层对应的JSP页面列表见表4-4-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 密码修改页面 | modifyPw.jsp | 用户进行密码修改 |

流程图如图4-4-1所示。



流程图4-4-1

### 4.4.2控制层

修改密码模块的控制层负责接受来自用户的密码修改，同时调用密码业务逻辑接口，将密码传递到业务逻辑层进行判定。等到业务逻辑层处理完毕后，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4-4-2所示。

表 4-4-2 登入模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 修改 | modifyPwAction.java | SUCCESS | mainFrame.jsp—修改成功，显示提示窗口。 |
| ERROR | modifyPw.jsp---修改失败，显示提示窗口。 |

在控制层中UploadAction.java的描述如下所示：

**package** com.buptsse.spm.action;

/\*\*

\* *@description* 实现资料上传页面Action

\*/

**public** **class** ModifyPwAction **extends** ActionSupport

{

**private** ModifyPw modifyPwd;

*@Resource*

**private** ModifyPwService modifyPw Service;

/\*实现各种属性的setter和getter方法\*/

/\*实现资料上传模块的控制层方法\*/

**public** String modify(){}

}

### 4.4.3业务逻辑层

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 修改 | ModifyPwService.java  ModifyPwServiceImpl.java | 调用ModifyPwDao | 返回给Action |

ModifyPwService.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** modifyPwService {

/\*修改密码\*/

**public** ModifyPw modify(String password);

}

ModifyPwServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** ModifyPwServiceImpl **implements** ModifyPwService {

*@Resource*

**Private** ModifyPwDao modifyPwDao;

*@Override*

**public** ModifyPw upload(String password);

}

### 4.4.4数据持久层

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 修改 | ModifyPwDao.java  ModifyPwDaoImpl.java | 调用 dodifyPw.Java | 对密码进行修改 |

ModifyPwDao.java

package com.buptsse.spm.dao;

**public** **interface** ModifyPwDao {

**public** ModifyPw modify (ModifyPw modifyPw); /\*修改密码\*/

}

### 4.4.5域模型层

|  |  |
| --- | --- |
| 域模型 | 描述 |
| ModifyPw.java | 对密码进行修改操作 |

ModifyPw.java主要属性与方法：

**package** com.buptsse.spm.domain;

**public** **class** ModifyPw **implements** Serializable{

**private** **String** passwo； /\*密码\*/

/\*编写对上述属性的*getter*与*setter*方法\*/

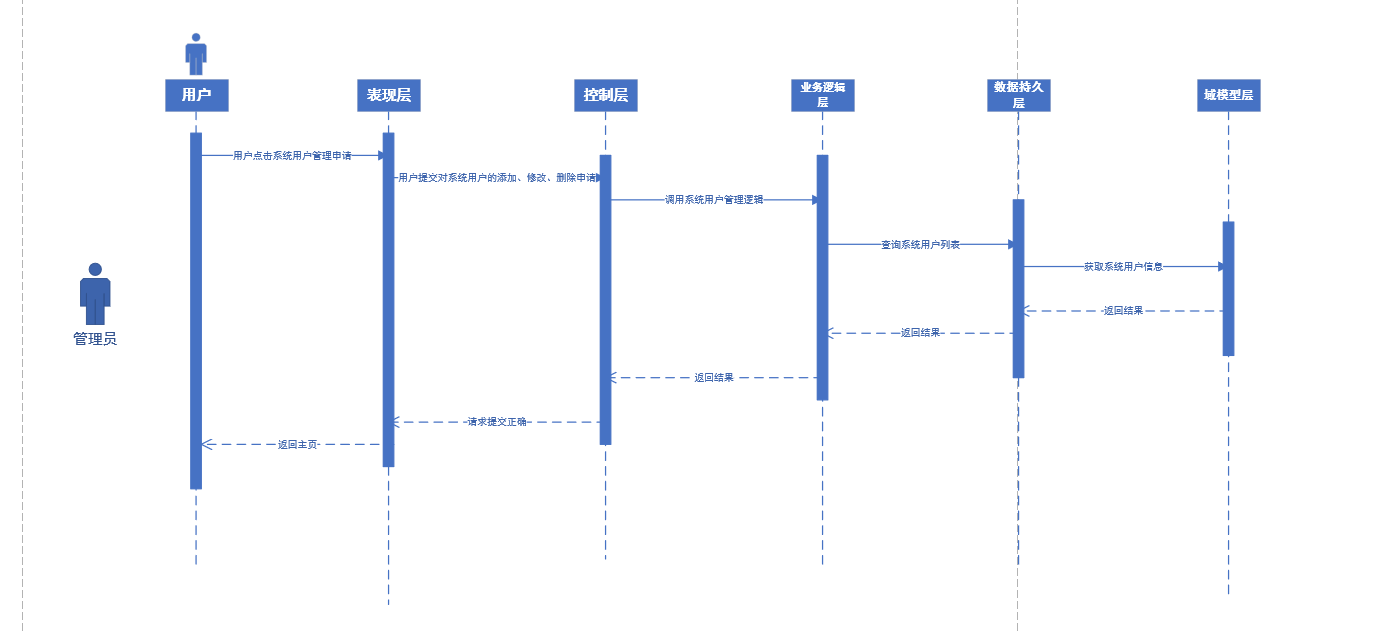
}

# 5. 管理端模块详细设计

管理端模块主要包括系统用户管理、课程信息管理模块、网站新闻管理模块。

## 5.1 系统用户管理详细设计模块

在系统用户管理模块时，系统内部的相应响应操作流程图如图5-1所示。



操作流程图5-1

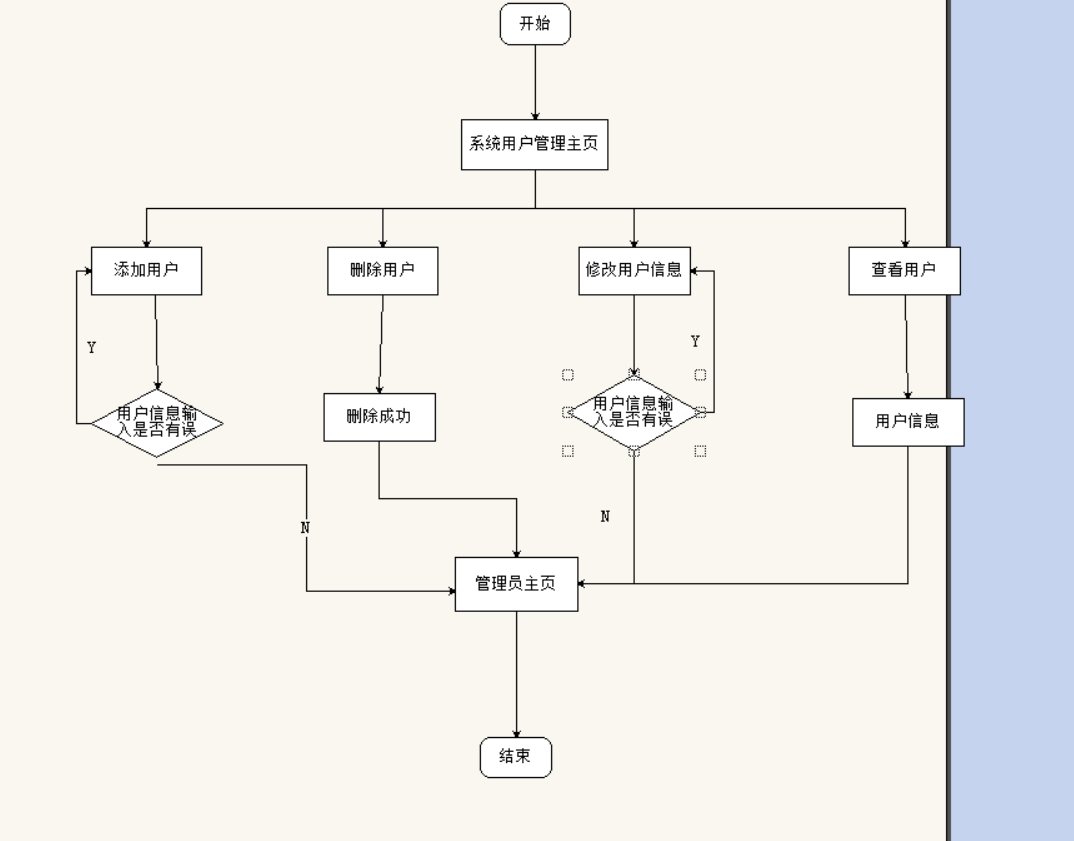
### 5.1.1表现层

系统用户模块的表现层主要是实现对系统用户的添加、删除、查询、修改功能。

表现层对应的JSP页面列表见表5-1-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 系统用户管理页面 | userMant.Jsp | 对系统用户进行管理 |

流程图如图5-1-1所示。



流程图5-1-1

### 5.1.2控制层

系统用户管理模块的控制层负责接受来自用户对系统用户管理的操作，同时调用系统用户管理业务逻辑接口，将用户账号、密码关键信息传递到业务逻辑层进行判定。等到业务逻辑层处理完毕后，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。系统用户管理模块控制层列表见表5-1-2所示。

表 5-1-2 系统用户管理模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 添加、  删除、  修改、  查看 | UserMantAction.java | SUCCESS | mainFrame.jsp—上传成功，显示提示窗口。 |
| ERROR | userMant.jsp---上传失败，显示提示窗口。 |

在控制层中UserMantAction.java的描述如下所示：

**package** com.buptsse.spm.action;

/\*\*

\*/

**public** **class** UserMantAction **extends** ActionSupport

{

**private** User user;

*@Resource*

**private** User Service user; Service;

/\*实现各种属性的setter和getter方法\*/

/\*实现资料上传模块的控制层方法\*/

**public** String add(){}

}

**public** String delect(){}

}

**public** String modify(){}

}

**public** String check(){}

}

### 5.1.3业务逻辑层

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 添加、  删除、  修改、  查看 | UserMantService.java  UserMantServiceImpl.java | 调用UserMantDao | 返回给Action |

UserMantService.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** userMantService {

**public** User user(int id,String userName,String password);

}

userMantServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** UserMantServiceImpl **implements** UserMantService {

*@Resource*

**Private** UserMantDao userMantDao;

*@Override*

**public** User user( int id,String userName,String password*);*

}

### 5.1.4数据持久层

系统用户管理模块的是数据持久层能对系统用户的管理。系统用户管理模块数据持久层列表见表4-3-4所示。

表 5-1-4 系统用户管理模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 添加、  删除、  修改、  查看 | IUserDao.java  UserDaoImpl.java | 调用 userMant.Java | 对系统用户进行管理 |

UserMantDao.java

package com.buptsse.spm.dao;

/\*\*

\* @description 用户信息持久层接口定义，包括用户增加，保存，查询

\*/

**public** **interface** IUserDao {

**public** User findUser(User user); /\*查找用户信息\*/

**public** List<User> searchUser(List<Object> choose); /\*查找用户信息\*/

**public** boolean deleteUser(String username); /\*删除用户信息\*/

**public** boolean addUser(User user); /\*增加用户信息\*/

}

### 5.1.5域模型层

UserMant.java主要属性与方法：

**package** com.buptsse.spm.domain;

**public** **class** UserMant **implements** Serializable{

**private** **String** id; /\*id识别\*/

**private** **String** userName; /\*姓名\*/

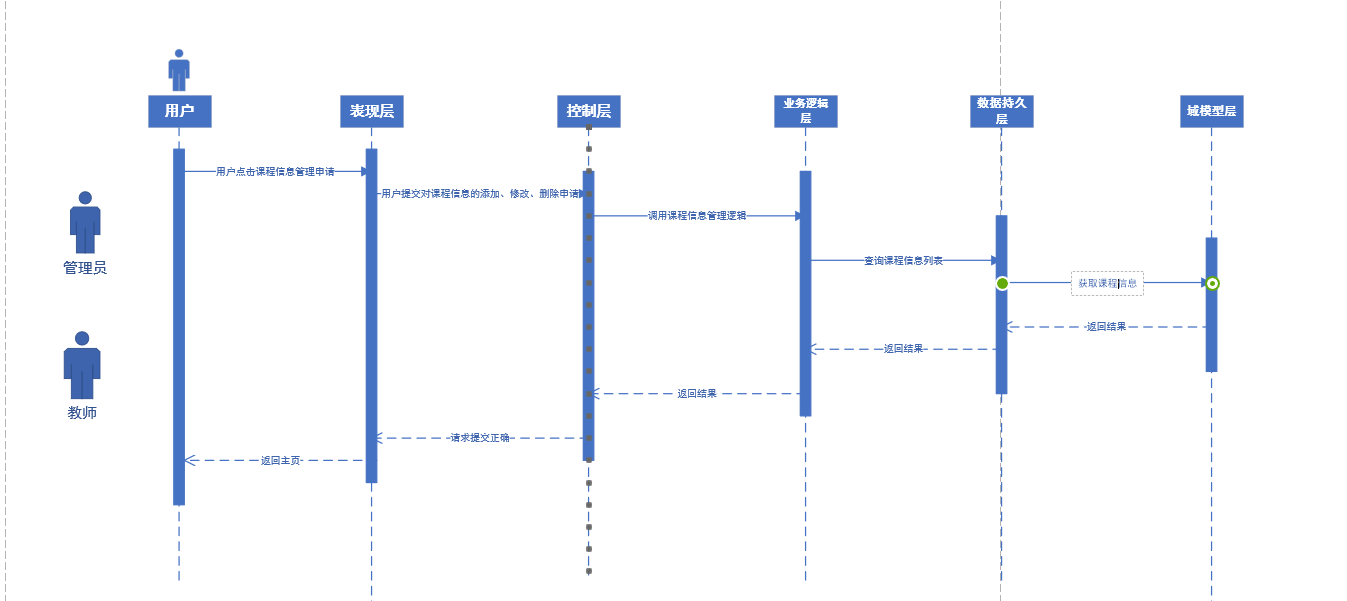
**private** **String** password; /\*密码\*/

/\*编写对上述各个属性的*getter*与*setter*方法\*/

}

## 5.2 课程信息管理模块详细设计

在课程信息管理模块时，系统内部的相应响应操作流程图如图5-2所示。



操作流程图如图5-2

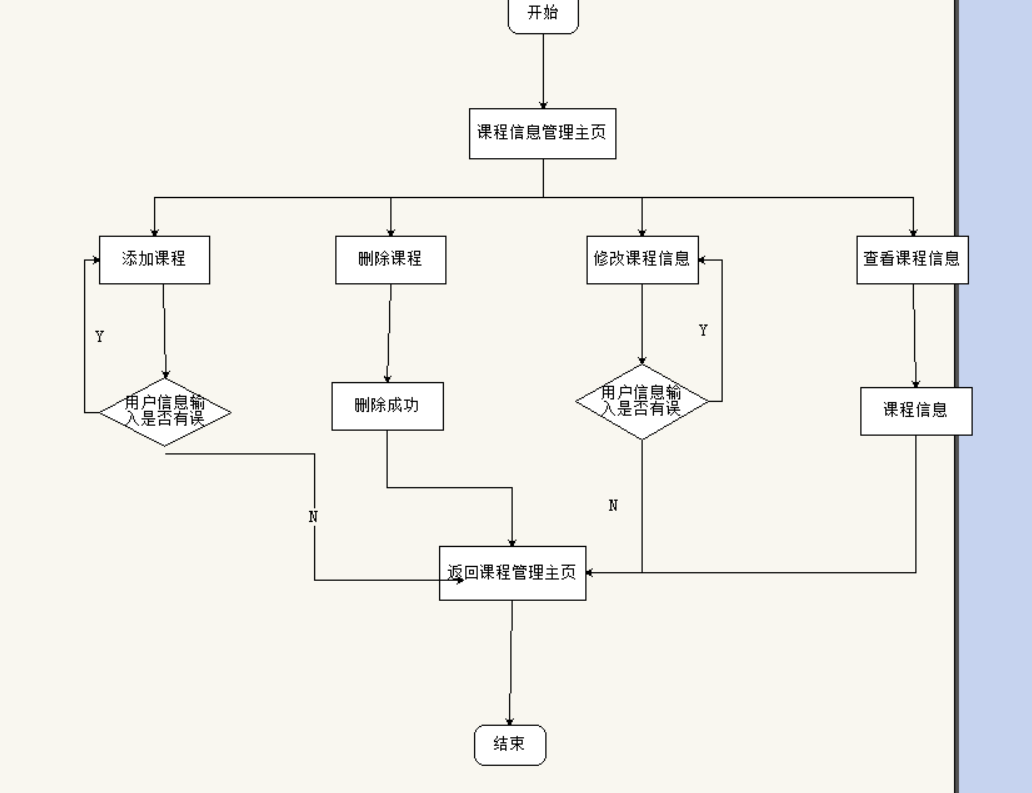
### 5.2.1表现层

课程信息管理模块的表现层主要是实现对课程信息的添加、删除、查询、修改功能。

表现层对应的JSP页面列表见表5-2-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 课程信息管理页面 | courseMant.jsp | 用户进行课程信息管理 |

流程图如图5-2-1所示。



流程图5-2-1

### 5.2.2控制层

课程信息管理模块的控制层负责接受来自用户的文件上传，同时调用资料课程信息管理业务逻辑接口，将课程id,课程名等关键信息传递到业务逻辑层进行判定。等到业务逻辑层处理完毕后，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。课程信息管理模块控制层列表见表5-2-2所示。

表 5-2-2 课程信息管理模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 添加、删除、修改、查看 | CourseMantAction.java | SUCCESS | mainFrame.jsp—添加、删除、修改、查看成功，显示提示窗口。 |
| ERROR | courseMant.jsp---添加、删除、修改、查看失败，显示提示窗口。 |

在控制层中CourseMantAction.java的描述如下所示：

**package** com.buptsse.spm.action;

/\*\*

\* *@description* 实现资料上传页面Action

\*/

**public** **class** CourseMantAction Action **extends** ActionSupport

{

**private** CourseMant courseMant;

*@Resource*

**private** CourseMant Service courseMant Service;

/\*实现各种属性的setter和getter方法\*/

/\*实现资料上传模块的控制层方法\*/

**public** String upload(){}

}

### 5.2.3业务逻辑层

CourseMantService.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** CourseMantService {

**public** CourseMant courseMant (int id,String courseName);

### 5.2.4数据持久层

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 添加、删除、修改、查看 | CourseMant Dao.java  CourseMant DaoImpl.java | 调用 courseMant Java | 对课程信息添加、删除、修改、查看 |

CourseMant Dao.java

package com.buptsse.spm.dao;

**public** **interface** CourseMantDao {

**public** CourseMantfind Course(Course course); /\*查找课程信息\*/

**public** boolean insert(Course course); /\*插入课程信息\*/

**public** boolean deleteString couserName);

/\*删除课程信息\*/

**public** boolean addCourse(Course course); /\*增加课程信息\*/

}

### 5.2.5域模型层

CourseMant.java主要属性与方法：

**package** com.buptsse.spm.domain;

**public** **class** CourseMant **implements** Serializable{

**private** **String** id; /\*id识别\*/

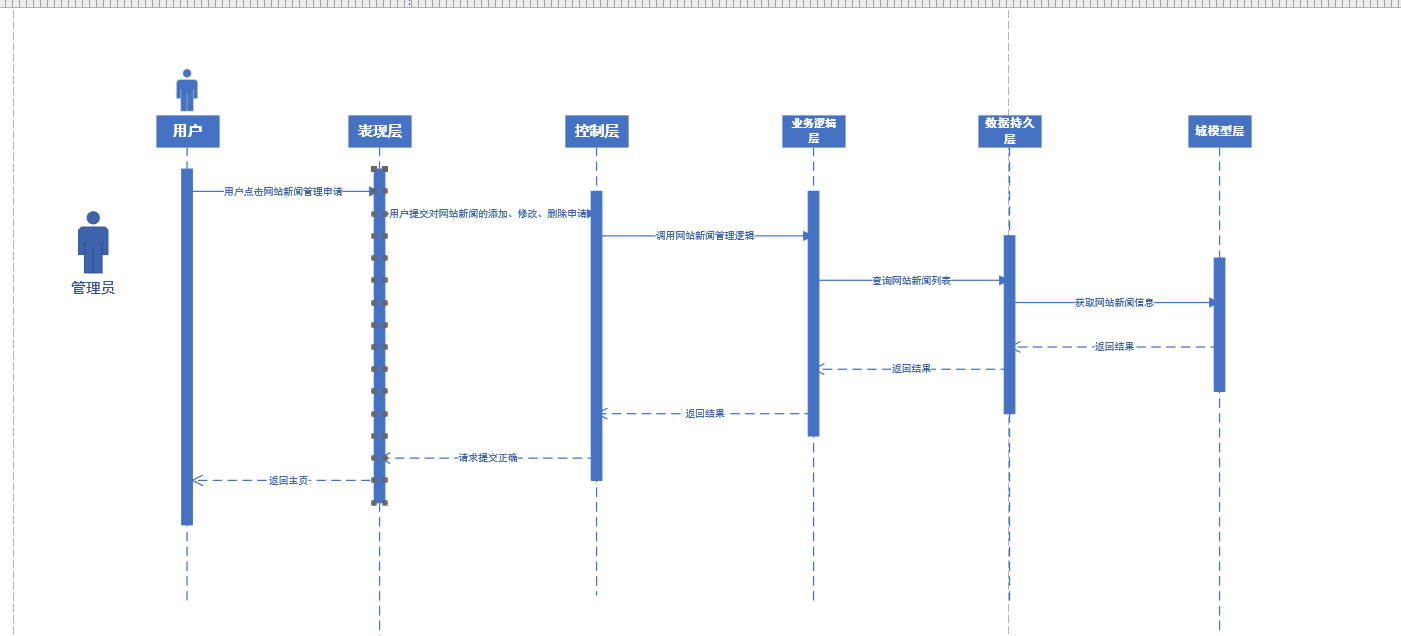
**private** **String** courseName; /\*课程名\*/

/\*编写对上述各个属性的*getter*与*setter*方法\*/

}

## 5.3 网站新闻管理模块详细设计

在网站新闻管理模块时，系统内部的相应响应操作流程图如图5-3所示。



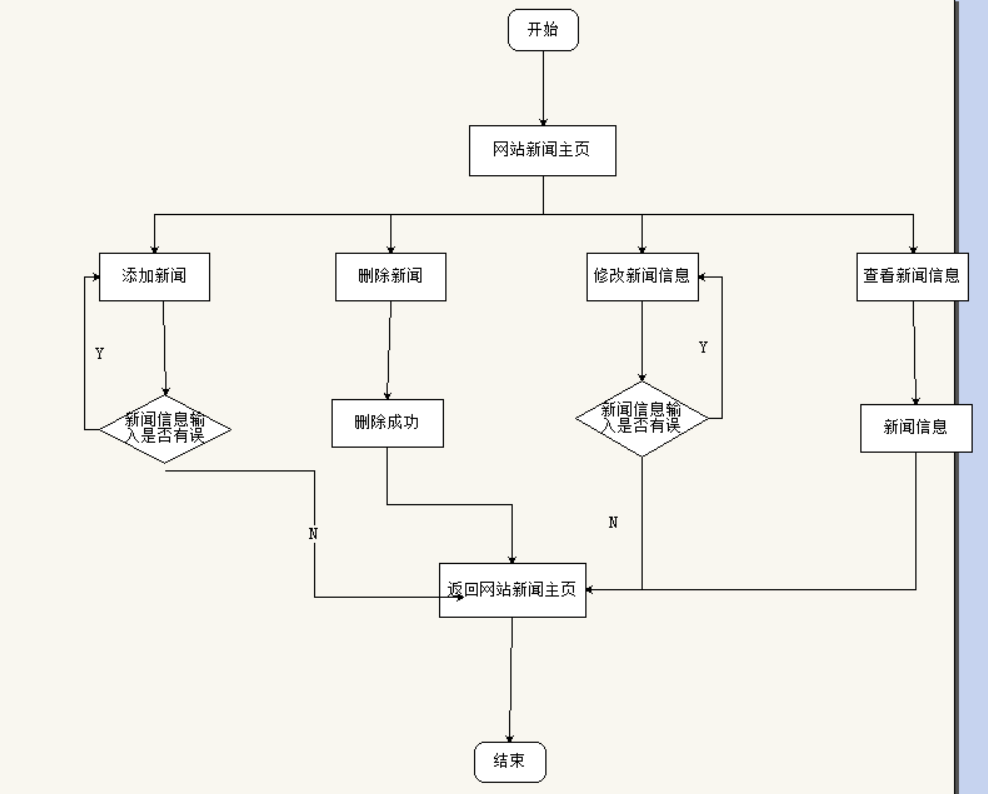
操作流程图5-3

### 5.3.1表现层

网站新闻管理模块的表现层主要是实现用户对网站新闻的添加、删除、查询、修改功能。表现层对应的JSP页面列表见表5-3-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 网站新闻管理页面 | newsMant.jsp | 用户对网站新闻进行管理 |

流程图如图5-3-1所示。



流程图5-3-1

### 5.3.2控制层

网站新闻管理模块的控制层负责接受来自用户的网站新闻管理，同时调用网站新闻管理业务逻辑接口，将新闻的id 新闻名等关键信息传递到业务逻辑层进行判定。等到业务逻辑层处理完毕后，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。网站新闻管理模块控制层列表见表5-3-2所示。

表 5-3-2 网站新闻管理模块控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 添加、  删除、  修改、  查看 | NewsMantAction.java | SUCCESS | mainFrame.jsp—添加、删除、修改、查看成功，显示提示窗口。 |
| ERROR | newsMant.jsp---添加、删除、修改、查看失败，显示提示窗口。 |

在控制层中NewsMantAction.java的描述如下所示：

**package** com.buptsse.spm.service;

**public** **interface** newsMant Service {

**public** newsMant newsMant (int id,StringnewsName);

}

**package** com.buptsse.spm.action;

/\*\*

\* *@description* 实现资料上传页面Action

\*/

**public** **class** NewsMantAction **extends** ActionSupport

{

**private** News news;

*@Resource*

**private** News Service news Service;

/\*实现各种属性的setter和getter方法\*/

/\*实现资料上传模块的控制层方法\*/

}

### 5.3.3业务逻辑层

newsMantServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** newsMant ServiceImpl **implements** UploadService {

*@Resource*

**private** NewsMant DaoNewsMant Dao;

*@Override*

**public** NewsMant newsMant (int id,String newsName);

}

### 5.3.4数据持久层

public interface NewsMantDao {

public NewsManfind NewsMan (News news);

### 5.3.5域模型层

CourseMant.java主要属性与方法：

package com.buptsse.spm.domain;

public class NewsMant implements Serializable{

private String id; /\*id识别\*/

private String newsName; /\*新闻名\*/

/\*编写对上述各个属性的getter与setter方法\*/

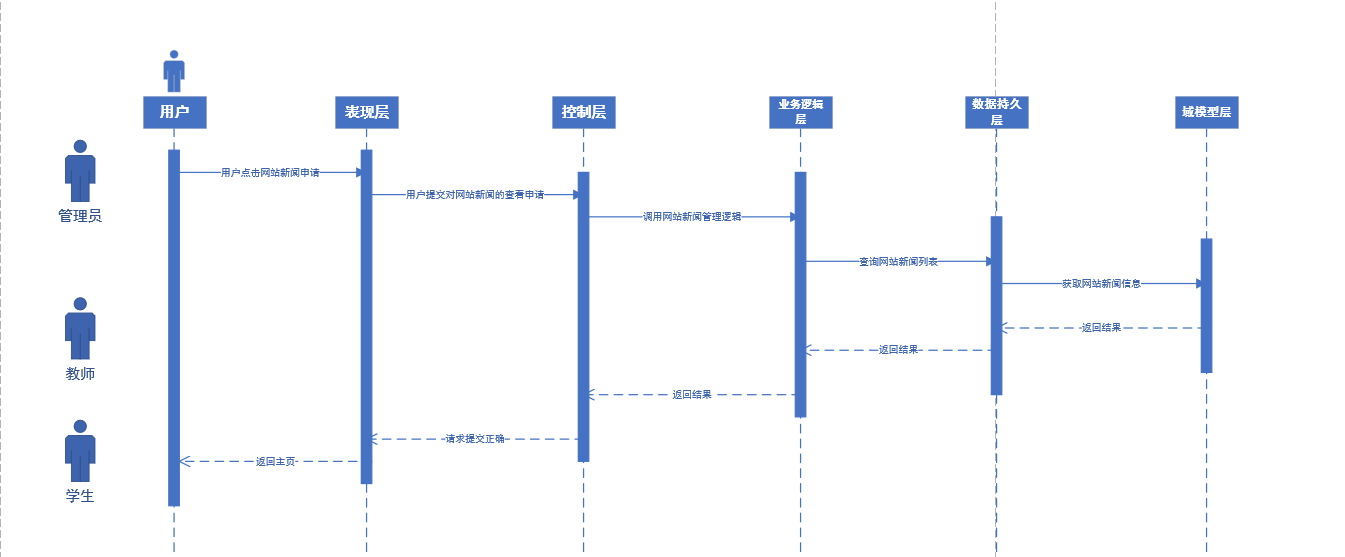
}

# 6. 公共部分模块详细设计

公共部分模块主要包括查看网站新闻、查看课程信息

## 6.1 查看网站新闻模块详细设计

在查看网站新闻模块时，系统内部的相应响应操作流程图如图6-1所示。



操作流程图如图6-1

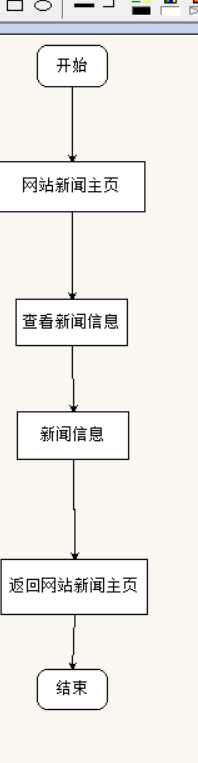
### 6.1.1表现层

查看网站新闻模块的表现层主要是实现用户查看网站新闻功能

表现层对应的JSP页面列表见表6-1-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 查看网站新闻页面 | cheakNews | 用户查看网站新闻 |

流程图如图6-1-1所示。



流程图6-1-1

### 6.1.2控制层

查看网站新闻模块的控制层负责接受来自用户的网站新闻查看，同时调用网站新闻业务逻辑接口，将新闻的Id 新闻名等关键信息传递到业务逻辑层进行判定。等到业务逻辑层处理完毕后，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。网站新闻查看模块控制层列表见表6-1-2所示。

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 查看 | CheckNewsAction.java | SUCCESS | mainFrame.jsp—查看成功，显示提示窗口。 |
| ERROR | CheckNews.jsp---查看失败，显示提示窗口。 |

表 6-1-2查看网站新闻控制层列表

在控制层中CheckNewsAction.java的描述如下所示：

**package** com.buptsse.spm.action;

**public** **class** CheckNewsAction **extends** ActionSupport

{

**private** Check check;

*@Resource*

**private** Check Service check Service;

/\*实现各种属性的setter和getter方法\*/

/\*实现资料上传模块的控制层方法\*/

**public** Stringcheck(){}

}

### 6.1.3业务逻辑层

CheckNews Service.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** CheckNews Service {

**public** CheckNews checkNews (int id,String newsName);

}

CheckNewsServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** CheckNews ServiceImpl **implements** CheckNewsService {

*@Resource*

**Private** CheckNewsDao checkNewsDao;

*@Override*

**public** Check check(int id,String newsName);

### 6.1.4数据持久层

package com.buptsse.spm.dao;

**public** **interface** CheckNewsrDao {

**public** CheckNews checkNews (CheckNews CheckNews);

}

### 6.1.5域模型层

**package** com.buptsse.spm.domain;

**public** **class** Check **implements** Serializable{

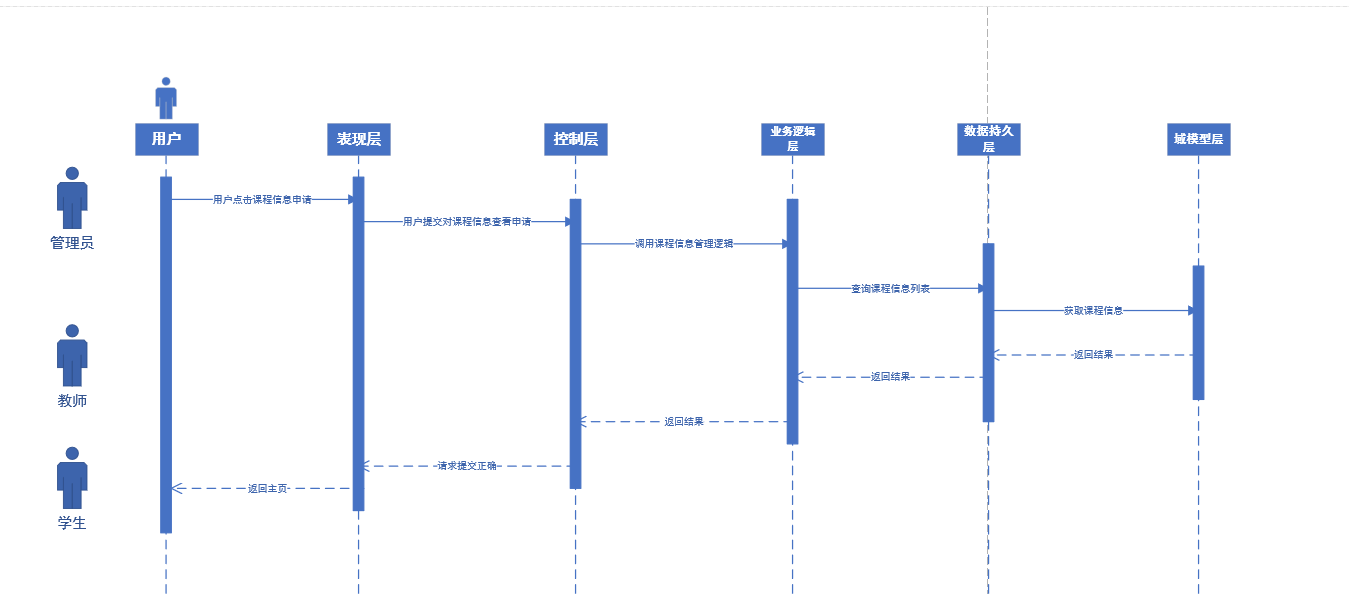
**private** **String** id; /\*id识别\*/

**private** **String** newsName; /\*新闻名\*/

}

## 6.2 查看课程信息模块详细设计

在查看课程信息模块时，系统内部的相应响应操作流程图如图6-2所示。



操作流程图如图6-2

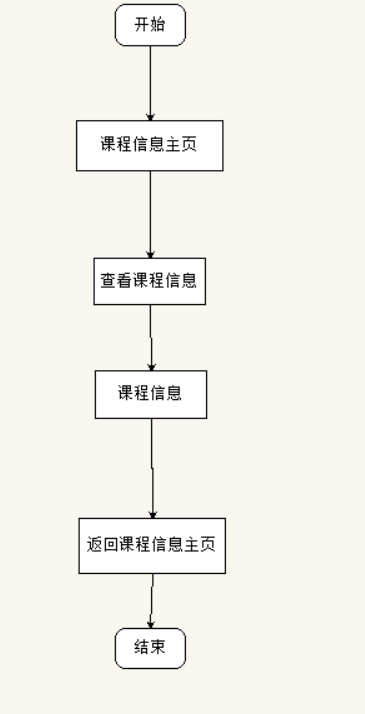
### 6.2.1表现层

查看课程信息模块的表现层主要是实现用户查看课程信息功能。

表现层对应的JSP页面列表见表6-2-1所示。

|  |  |  |
| --- | --- | --- |
| 界面 | JSP | 功能描述 |
| 查看课程信息页面 | CheckCourse.jsp | 用户查看课程信息 |

流程图如图6-2-1所示。



流程图6-2-1

### 6.2.2控制层

查看课程信息模块的控制层负责接受来自用户的课程信息查看，同时调用课程信息业务逻辑接口，将课程的Id 课程名等关键信息传递到业务逻辑层进行判定。等到业务逻辑层处理完毕后，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。查看课程信息模块控制层列表见表6-2-2所示。

表 6-2-2查看课程信息控制层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Action | 转移说明 | 出口 |
| 查看 | CheckCourseAction.java | SUCCESS | mainFrame.jsp—查看成功，显示提示窗口。 |
| ERROR | CheckCourse.jsp---查看失败，显示提示窗口。 |

在控制层中CheckCourseAction.java的描述如下所示：

**package** com.buptsse.spm.action;

**public** **class** CheckNewsAction **extends** ActionSupport

{

**private** Check check;

*@Resource*

**private** Check Service check Service;

/\*实现各种属性的setter和getter方法\*/

/\*实现资料上传模块的控制层方法\*/

**public** String check (){}

}

### 6.2.3业务逻辑层

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | Service | 调用说明 | 出口 |
| 查看 | CheckCourseService.java  CheckCourseServiceImpl.java | 调用CheckCourseDao | 返回给Action |

CheckCourseService.java接口主要方法：

**package** com.buptsse.spm.service;

**public** **interface** checkCourseService {

**public** Check check(int id,String courseName);

}

CheckCourseServiceImpl.java实现了CheckCourseService.java接口， 同时需要调用数据持久层的CheckCourseDao.Java，利用@Resource来实现对数据持久层接口的调用。

CheckCourseServiceImpl.java主要实现属性与方法:

**package** com.buptsse.spm.service.impl;

*@Transactional*

*@Service*

**public** **class** CheckCourseServiceImpl **implements** CheckCourseService {

*@Resource*

**Private** CheckCourseDao CheckCourseDao;

*@Override*

**public** Check check(int id,String courseName);

### 6.2.4数据持久层

课程信息查看模块的是数据持久层能对课程信息的查看。课程信息查看数据持久层列表见表6-2-4所示。

表 6-2-4 课程信息查看模块数据持久层列表

|  |  |  |  |
| --- | --- | --- | --- |
| 事件 | DAO | 调用数据模型 | 说明 |
| 查看 | CheckCourse Dao.java  CheckCourse DaoImpl.java | 调用 CheckCourse Java | 对课程信息进行查看 |

package com.buptsse.spm.dao;

**public** **interface** CheckCoureseDao {

**public** CheckCourese checkCourese (CheckCourese CheckCourese);

}

### 6.2.5域模型层

|  |  |
| --- | --- |
| 域模型 | 描述 |
| CheckCourse.java | 对课程信息进行查看操作 |

CheckCourse.java主要属性与方法：

**package** com.buptsse.spm.domain;

**public** **class** Check **implements** Serializable{

**private** **String** id; /\*id识别\*/

**private** **String** CourseName; /\*课程名\*/

}