目录

[1. 引言](#_Toc32020_WPSOffice_Level1) [5](#_Toc32020_WPSOffice_Level1)

[1.1. 编写目的](#_Toc18178_WPSOffice_Level2) [5](#_Toc18178_WPSOffice_Level2)

[1.2. 项目背景](#_Toc4401_WPSOffice_Level2) [5](#_Toc4401_WPSOffice_Level2)

[1.3. 定义](#_Toc26987_WPSOffice_Level2) [5](#_Toc26987_WPSOffice_Level2)

[1.4. 参考资料](#_Toc25627_WPSOffice_Level2) [5](#_Toc25627_WPSOffice_Level2)

[2. 总体设计](#_Toc18178_WPSOffice_Level1) [5](#_Toc18178_WPSOffice_Level1)

[2.1. 需求概述](#_Toc13027_WPSOffice_Level2) [5](#_Toc13027_WPSOffice_Level2)

[2.2. 系统结构](#_Toc28468_WPSOffice_Level2) [6](#_Toc28468_WPSOffice_Level2)

[3. 程序描述](#_Toc4401_WPSOffice_Level1) [6](#_Toc4401_WPSOffice_Level1)

[3.1. 登录模块](#_Toc20660_WPSOffice_Level2) [6](#_Toc20660_WPSOffice_Level2)

[3.1.1. 效果图](#_Toc18178_WPSOffice_Level3) [6](#_Toc18178_WPSOffice_Level3)

[3.1.2. 功能流程图](#_Toc4401_WPSOffice_Level3) [7](#_Toc4401_WPSOffice_Level3)

[3.1.3. 功能描述](#_Toc26987_WPSOffice_Level3) [7](#_Toc26987_WPSOffice_Level3)

[3.1.4. 界面设计](#_Toc25627_WPSOffice_Level3) [7](#_Toc25627_WPSOffice_Level3)

[3.1.5. 内部逻辑的实现](#_Toc13027_WPSOffice_Level3) [7](#_Toc13027_WPSOffice_Level3)

[3.2. 学生考试模块](#_Toc30601_WPSOffice_Level2) [8](#_Toc30601_WPSOffice_Level2)

[3.2.1. 效果图](#_Toc28468_WPSOffice_Level3) [8](#_Toc28468_WPSOffice_Level3)

[3.2.2. 功能流程图](#_Toc20660_WPSOffice_Level3) [8](#_Toc20660_WPSOffice_Level3)

[3.2.3. 功能描述](#_Toc30601_WPSOffice_Level3) [9](#_Toc30601_WPSOffice_Level3)

[3.2.4. 界面设计](#_Toc13567_WPSOffice_Level3) [9](#_Toc13567_WPSOffice_Level3)

[3.2.5. 内部逻辑的实现](#_Toc27379_WPSOffice_Level3) [9](#_Toc27379_WPSOffice_Level3)

[3.2.6. 存储分配](#_Toc12676_WPSOffice_Level3) [10](#_Toc12676_WPSOffice_Level3)

[3.3. 学生密码修改模块](#_Toc13567_WPSOffice_Level2) [10](#_Toc13567_WPSOffice_Level2)

[3.3.1. 效果图](#_Toc18998_WPSOffice_Level3) [10](#_Toc18998_WPSOffice_Level3)

[3.3.2. 功能流程图](#_Toc2523_WPSOffice_Level3) [10](#_Toc2523_WPSOffice_Level3)

[3.3.3. 功能描述](#_Toc237_WPSOffice_Level3) [11](#_Toc237_WPSOffice_Level3)

[3.3.4. 界面设计](#_Toc16427_WPSOffice_Level3) [11](#_Toc16427_WPSOffice_Level3)

[3.3.5. 内部逻辑的实现](#_Toc28927_WPSOffice_Level3) [12](#_Toc28927_WPSOffice_Level3)

[3.3.6. 存储分配](#_Toc22265_WPSOffice_Level3) [12](#_Toc22265_WPSOffice_Level3)

[3.4. 教师考试概况模块](#_Toc28927_WPSOffice_Level2) [12](#_Toc28927_WPSOffice_Level2)

[3.4.1. 效果图](#_Toc18282_WPSOffice_Level3) [12](#_Toc18282_WPSOffice_Level3)

[3.4.2. 功能流程图](#_Toc3173_WPSOffice_Level3) [13](#_Toc3173_WPSOffice_Level3)

[3.4.3. 功能描述](#_Toc16954_WPSOffice_Level3) [13](#_Toc16954_WPSOffice_Level3)

[3.4.4. 界面设计](#_Toc31581_WPSOffice_Level3) [13](#_Toc31581_WPSOffice_Level3)

[3.4.5. 内部逻辑的实现](#_Toc17733_WPSOffice_Level3) [13](#_Toc17733_WPSOffice_Level3)

[3.5. 教师课程管理模块](#_Toc22265_WPSOffice_Level2) [14](#_Toc22265_WPSOffice_Level2)

[3.5.1. 课程信息](#_Toc14344_WPSOffice_Level3) [14](#_Toc14344_WPSOffice_Level3)

[3.5.2. 题库管理](#_Toc7419_WPSOffice_Level3) [16](#_Toc7419_WPSOffice_Level3)

[3.6. 教师考核管理模块](#_Toc18282_WPSOffice_Level2) [19](#_Toc18282_WPSOffice_Level2)

[3.6.1. 考试管理](#_Toc27231_WPSOffice_Level3) [19](#_Toc27231_WPSOffice_Level3)

[3.6.2. 成员管理](#_Toc12720_WPSOffice_Level3) [20](#_Toc12720_WPSOffice_Level3)

[3.7. 教师成绩管理模块](#_Toc3173_WPSOffice_Level2) [22](#_Toc3173_WPSOffice_Level2)

[3.7.1. 考试成绩](#_Toc8369_WPSOffice_Level3) [22](#_Toc8369_WPSOffice_Level3)

[3.7.2. 功能流程图](#_Toc19069_WPSOffice_Level3) [23](#_Toc19069_WPSOffice_Level3)

[3.7.3. 功能描述](#_Toc22429_WPSOffice_Level3) [23](#_Toc22429_WPSOffice_Level3)

[3.7.4. 界面设计](#_Toc997_WPSOffice_Level3) [23](#_Toc997_WPSOffice_Level3)

[3.7.5. 内部逻辑实现](#_Toc27272_WPSOffice_Level3) [23](#_Toc27272_WPSOffice_Level3)

[3.8. 成绩明细](#_Toc16954_WPSOffice_Level2) [24](#_Toc16954_WPSOffice_Level2)

[3.8.1.1. 效果图](#_Toc219_WPSOffice_Level3) [24](#_Toc219_WPSOffice_Level3)

[3.8.1.2. 功能流程图](#_Toc31001_WPSOffice_Level3) [24](#_Toc31001_WPSOffice_Level3)

[3.8.1.3. 功能描述](#_Toc20715_WPSOffice_Level3) [25](#_Toc20715_WPSOffice_Level3)

[3.8.1.4. 界面设计](#_Toc28996_WPSOffice_Level3) [25](#_Toc28996_WPSOffice_Level3)

[3.8.1.5. 内部逻辑实现](#_Toc7297_WPSOffice_Level3) [25](#_Toc7297_WPSOffice_Level3)

[3.9. 管理员学生管理模块](#_Toc31581_WPSOffice_Level2) [26](#_Toc31581_WPSOffice_Level2)

[3.9.1. 效果图](#_Toc10976_WPSOffice_Level3) [26](#_Toc10976_WPSOffice_Level3)

[3.9.2. 功能流程图](#_Toc28847_WPSOffice_Level3) [27](#_Toc28847_WPSOffice_Level3)

[3.9.3. 功能描述](#_Toc10478_WPSOffice_Level3) [29](#_Toc10478_WPSOffice_Level3)

[3.9.4. 界面设计](#_Toc3552_WPSOffice_Level3) [29](#_Toc3552_WPSOffice_Level3)

[3.9.5. 内部逻辑实现](#_Toc30186_WPSOffice_Level3) [29](#_Toc30186_WPSOffice_Level3)

[3.9.6. 存储分配](#_Toc31062_WPSOffice_Level3) [30](#_Toc31062_WPSOffice_Level3)

[3.10. 管理员教师管理模块](#_Toc17733_WPSOffice_Level2) [30](#_Toc17733_WPSOffice_Level2)

[3.10.1. 效果图](#_Toc31719_WPSOffice_Level3) [30](#_Toc31719_WPSOffice_Level3)

[3.10.2. 功能流程图](#_Toc20327_WPSOffice_Level3) [31](#_Toc20327_WPSOffice_Level3)

[3.10.3. 功能描述](#_Toc32322_WPSOffice_Level3) [33](#_Toc32322_WPSOffice_Level3)

[3.10.4. 界面设计](#_Toc1234_WPSOffice_Level3) [33](#_Toc1234_WPSOffice_Level3)

[3.10.5. 内部逻辑的实现](#_Toc18988_WPSOffice_Level3) [34](#_Toc18988_WPSOffice_Level3)

[3.10.6. 存储分配](#_Toc30638_WPSOffice_Level3) [34](#_Toc30638_WPSOffice_Level3)

[3.11. 管理员班级管理模块](#_Toc14344_WPSOffice_Level2) [34](#_Toc14344_WPSOffice_Level2)

[3.11.1. 效果图](#_Toc7270_WPSOffice_Level3) [34](#_Toc7270_WPSOffice_Level3)

[3.11.2. 功能流程图](#_Toc29875_WPSOffice_Level3) [36](#_Toc29875_WPSOffice_Level3)

[3.11.3. 功能描述](#_Toc28121_WPSOffice_Level3) [37](#_Toc28121_WPSOffice_Level3)

[3.11.4. 界面设计](#_Toc15348_WPSOffice_Level3) [37](#_Toc15348_WPSOffice_Level3)

[3.11.5. 内部逻辑的实现](#_Toc9713_WPSOffice_Level3) [37](#_Toc9713_WPSOffice_Level3)

[3.11.6. 存储分配](#_Toc10304_WPSOffice_Level3) [38](#_Toc10304_WPSOffice_Level3)

# 引言

## 编写目的

随着计算机技术的飞速发展和高等教育体制改革的不断深入，传统教育管理方法、手段以及工作效率已不能适应新的发展需要，无法很好地完成教学管理工作。提高教学管理水平的主要途径是更新管理者的思想，增强对管理活动的科学认识。鉴于这种情况，我们开发一套在线考试系统，旨在为管理员、学生和教师提供了考试的平台，提供了一个更好的交流平台，真正实现网络的互动，功能比较齐全，基本上能满足学生和老师的要求。

## 项目背景

在线考试系统就是实现用户不管身处何地，只要可以使用网络浏览器 ，就可通过网络登录在线考试系统，参加在线考试，该网站能够较好的满足考试用户的基本要求，具有较高的实用价值。

## 定义

•MySQL ：数据库设计软件

•PL/SQL：数据库管理软件

•Windows：运行环境

•Java，：软件开发语言

•Myeclipse ,SVN：开发工具

## 参考资料

《软件工程导论——第5版》 张海藩编著 清华大学出版社

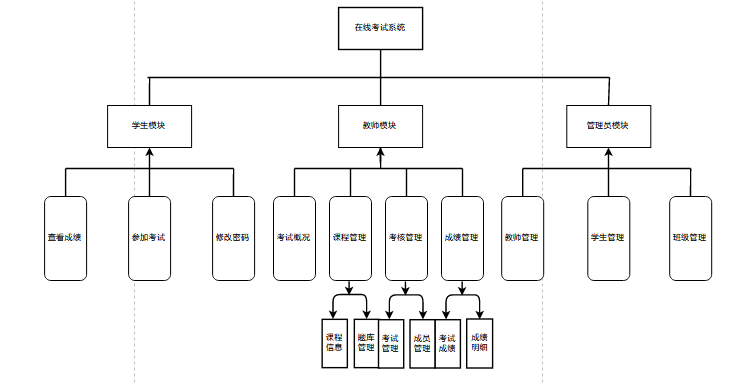
《java编程思想》 Bruce Eckel著 机械工业出版社

# 总体设计

## 需求概述

按照需求分析文档中的规格要求，使用登录信息进行登录、修改、添加，使得信息传递准确、流畅。同时，系统最大限度地实现易安装，易维护性，易操作性，运行稳定，安全可靠。

## 系统结构



# 程序描述

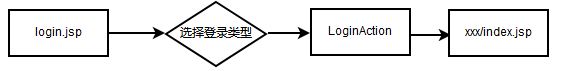
## 登录模块

### 效果图



### 功能流程图

功能流程图如下图所示：



需要说明的问题：

1. login.jsp提交的数据是用户名、密码及登录类型。
2. 当勾选“下次自动登录”，会记住密码。

### 功能描述

* 1. 功能类型：查询数据
  2. 功能描述：提供学生，教师，管理员用户进行登录的平台
  3. 操作权限：学生，教师，管理员

### 界面设计

* + 1. 数据要求

1）功能类型：数据查询

2）数据描述：

页面显示录入字段如下：

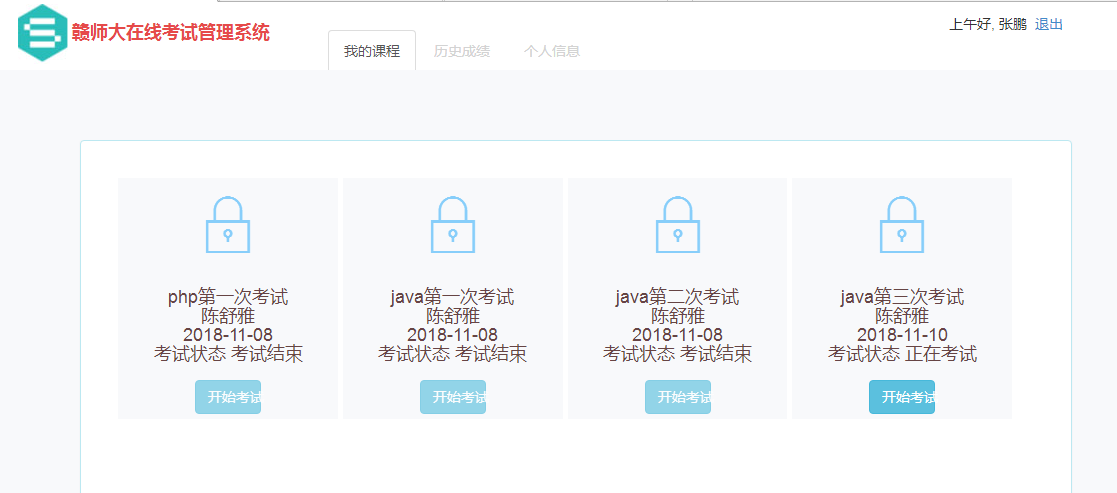
|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| xxxId | int | 学生/老师/管理员编号 |
| xxxpwd | varchar | 密码 |

### 内部逻辑的实现

（1）接收登录表单提交的学号、密码及登录类型，通过调用LoginAction中的方法，获取登录用户的实体对象，如果返回的对象为NULL，则返回并提示“登录失败，请重新登录”，如果不为NULL，则将用户对象保存到SESSION（session.setAttribute(“xxx”)）,当勾选“下次自动登录”时，同时保存到COOKIE(COOKIE中保存用户名membername和密码memberpwd),最后跳转至index.jsp。

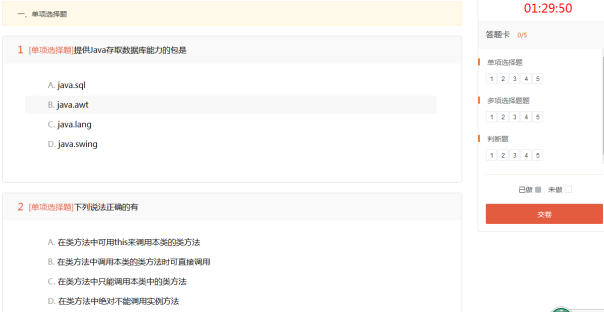
## 学生考试模块

### 效果图



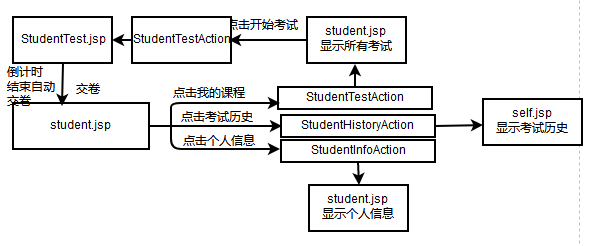






### 功能流程图

功能流程图如下图所示：



需要说明的问题：

（1）已经结束的考试“考试按钮”不可点击。

### 功能描述

1. 功能类型：查找数据，添加数据
2. 功能描述：查找课程，考试历史和个人信息，学生填写试卷
3. 操作权限：学生

### 界面设计

（1）数据要求

1）功能类型：添加数据

2）数据描述：

页面显示录入字段如下：

学生填写试题答案。

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| StudentAnswer | Varchar | 学生答案 |

### 内部逻辑的实现

1）点击“我的课程”，student.jsp将学生Id传递给StudentTestAction，StudentTestAction进行查询，将结果返回给student.jsp，student.jsp显示相应数据。

2）点击“开始考试”student.jsp将考试Id传递给StudentTestAction，StudentTestAction进行查询，将相应试卷信息传递给StudentTest.jsp，StudentTest.jsp显示试卷。点击提交试卷或倒计时结束StudentTest.jsp将考生客观题成绩计算，传递给StudentTestAction，StudentTestAction进行相应数据处理。

3）点击“历史成绩”，student.jsp将学生Id传递给StudentHistoryAction，StudentHistoryAction进行查询，将结果返回给student.jsp，student.jsp显示相应数据。

4）点击“个人信息”，student.jsp将学生Id传递给StudentInfoAction，StudentInfoAction进行查询，将结果返回给self.jsp，self.jsp显示相应数据。

### 存储分配

考生答案表（resultinfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| resultID | Int 主键 |  |
| paperID | Int | 试卷Id |
| studentId | Int | 学生ID |
| titleCat | int | 题目分类 |
| titleId | int | 题目 |
| StudentAnswer | Varchar | 学生答案 |
| CorrectState | Int | 是否批改，0未批改 |

学生状态表（studentstate）

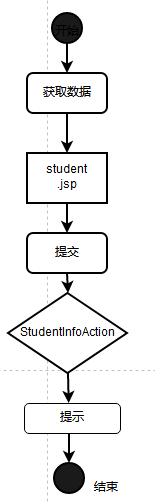
|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| stateId | Int 主键 |  |
| paperId | Int | 试卷ID |
| studentId | Int | 学生ID |
| stateFlag | Int | 学生状态（0：未考试，1：正在考试，2：结束考试） |

## 学生密码修改模块

### 效果图

### 功能流程图

功能流程图如下图所示：



需要说明的问题：

（1）点击登录密码“\*\*\*\*\*\*”处，会出现一个输入框，输入密码，点击输入框外提交。

### 功能描述

（1）功能类型：更新数据

（2）功能描述：用户修改密码

（3）操作权限：学生

### 界面设计

（1）数据要求

1）功能类型：查看数据

2）数据描述：

页面显示录入字段如下：

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| StudentPwd | Varchar | 学生密码 |

### 内部逻辑的实现

提交修改后密码，获取修改结果信息，提示用户

### 存储分配

学生表（studentinfo）

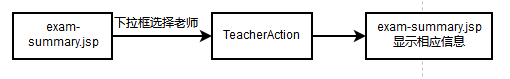
|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| studentId | Int 主键 |  |
| studentName | varchar | 学生姓名 |
| studentPwd | varchar | 密码 |
| classId | Int | 班级ID |
| birthday | date | 出生日期 |
| school | varchar | 所在学校 |
| address | varchar | 家庭住址 |
| phone | varchar | 联系电话 |

## 教师考试概况模块

### 效果图



### 功能流程图



需要说明的问题：

1）exam-summary.jsp默认显示所有考试信息。

### 功能描述

* 1. 功能类型：查询当前发布的考试、考试状态。
  2. 功能描述：通过点击考试概况来查看教师发布的考试、考试状态。
  3. 操作权限：教师

### 界面设计

（1）数据要求

1）功能类型：查询当前教师考试、考试状态及参考人数

2）数据描述：

页面显示录入字段如下：

1. 下拉框选择老师；

2）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| paperName | varchar | 试卷名称 |

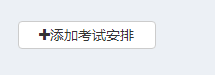
### 内部逻辑的实现

1. 默认进入exam-summary.jsp时，Exam\_summaryAction将所有考试信息传递给exam-summary.jsp，exam-summary.jsp将信息显示在页面。
2. 下拉框选择老师，exam-summary.jsp将老师Id传递给Exam\_summaryAction，Exam\_summaryAction将相应信息回应给exam-summary.jsp，exam-summary.jsp将信息显示在页面。
3. 文本框输入考试名称进行查询，exam-summary.jsp将考试名称传递给Exam\_summaryAction，Exam\_summaryAction将相应信息回应给exam-summary.jsp，exam-summary.jsp将信息显示在页面。

## 教师课程管理模块

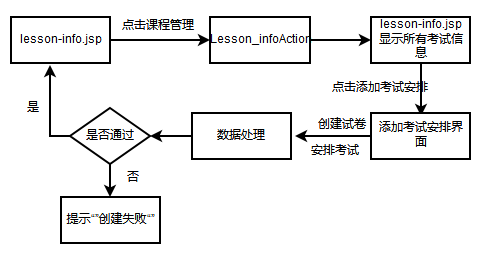
### 课程信息

#### 效果图





#### 功能流程图



需要说明的问题：

1）页面默认显示所有的课程信息。

#### 功能描述

1）功能类型：查询数据、添加数据

2）功能描述：显示课程信息。

3）操作权限：教师

#### 界面设计

（1）数据要求

1）功能类型：查询数据

2）数据描述：

页面显示录入字段如下：

（1）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| paperName | varchar | 试卷名称 |

（2）

下拉框选择所属课程，考试试卷；

选择框选择考试班级。

#### 内部逻辑

1）默认进入lesson-info.jsp时，Lesson\_infoAction将所有课程信息传递给lesson-info.jsp，lesson-info.jsp将信息显示在页面。

2）点击“添加考试安排”，页面刷新，输入试卷名称和选择考试课程lesson-info.jsp将这些信息传递给Lesson\_infoAction，Lesson\_infoAction进行数据处理并返回处理结果给lesson-info.jsp，lesson-info.jsp显示相应处理结果。

3）点击“添加考试安排”，页面刷新，选择考试试卷和考试班级lesson-info.jsp将这些信息传递给Lesson\_infoAction，Lesson\_infoAction进行数据处理并返回处理结果给lesson-info.jsp，lesson-info.jsp显示相应处理结果。

#### 存储分配

考试安排表（examplaninfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| examplanId | Int 主键 |  |
| paperId | Int | 试卷ID |
| classId | Int | 班级ID |
| StartTime | datatime | 开始时间 |
| OverTime | datatime | 结束时间 |
| Examstate | Int | 考试状态（0：未开考 1：已结束 2:正在进行） |

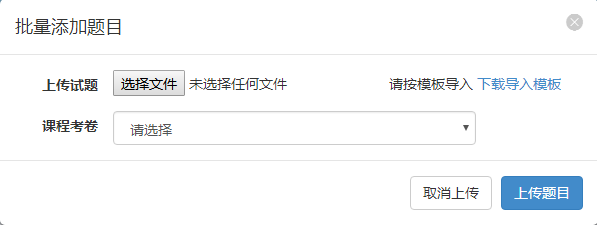
试卷表（paperinfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| PaperId | Int 主键 |  |
| paperName | varchar | 试卷名称 |
| CourseId | Int | 课程ID |
| isAdd | int | 0否，1是 |

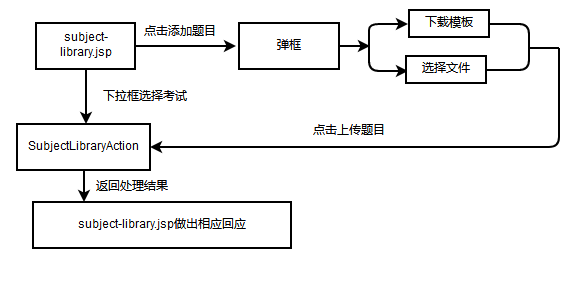
### 题库管理

#### 效果图





#### 功能流程图



#### 功能描述

1）功能类型：查询数据、添加数据

2）功能描述：显示试题信息。

3）操作权限：教师

#### 界面设计

（1）数据要求

1）功能类型：查询数据

2）数据描述：

页面显示录入字段如下：

1. 下拉框选择考试。
2. 文件选择框选择文件，下拉框选择所属课程。

#### 内部逻辑

1）默认进入subject-library.jsp时，SubjectLibraryAction将当前教师所发布的所有题目信息传递给lesson-info.jsp，subject-library.jsp将信息显示在页面。

2）点击“添加题目”，跳出弹框，选择文件和所属课程，subject-library.jsp将这些信息传递给SubjectLibraryAction，SubjectLibraryAction进行数据处理并返回处理结果给subject-library.jsp，subject-library.jsp显示相应处理结果。

3）下拉框选择考试，，subject-library.jsp将考试id传递给SubjectLibraryAction，SubjectLibraryAction进行数据处理并返回处理结果给subject-library.jsp，subject-library.jsp显示考试所有题目。

#### 存储分配

题目表（titleinfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| titleId | Int 主键 |  |
| titleCat | Int | 题目分类 |
| titleStem | Varchar | 题干 |
| optionA | varchar | 选项A |
| optionB | varchar | 选项B |
| optionC | varchar | 选项C |
| optionD | varchar | 选项D |
| answerRight | varchar | 正确答案 |
| answerScore | Int | 答案分数 |
| analyze | Varchar | 试题分析 |
| paperId | Int | 试卷ID |

## 教师考核管理模块

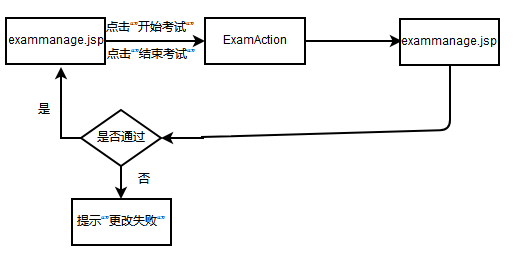
### 考试管理

#### 效果图



#### 功能流程图

功能流程图如下图所示：



需要说明的问题：

在考试管理页面页面中点击“开始考试”或“结束考试”按钮时改变考试状态

#### 功能描述

* 1. 功能类型：更新数据
  2. 功能描述：实现对考试状态的改变
  3. 操作权限：教师

#### 界面设计

* + 1. 数据要求

1. 功能类型：更新数据
2. 数据描述：

点击按钮对数据进行更新。

#### 内部逻辑实现

（1）点击“开始考试”或“结束考试”，Exam\_manageAction根据exam-manage.jsp传递的考试ID进行查询，并对考试状态进行更改，且返回处理结果给exam-manage.jsp，exam-manage.jsp显示处理结果。

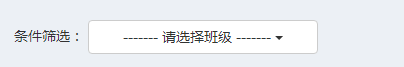
#### 存储分配

考试安排表（examplaninfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| examplanId | Int 主键 |  |
| paperId | Int | 试卷ID |
| classId | Int | 班级ID |
| StartTime | datatime | 开始时间 |
| OverTime | datatime | 结束时间 |
| Examstate | Int | 考试状态（0：未开考 1：已结束 2:正在进行） |

### 成员管理

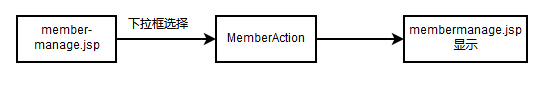
#### 效果图





#### 功能流程图

功能流程图如下图所示。



需要说明的问题

可以选择“班级”分类对考生的答题状态进行分类查询。

#### 功能描述

（1）功能类型：查询数据

（2）功能描述：查询考生的考试状态

（3）操作权限：教师

#### 界面设计

1. 数据要求

1）功能类型：查询数据

2）数据描述：

页面显示录入字段如下：

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| className | varchar | 班级名称 |

#### 内部逻辑实现

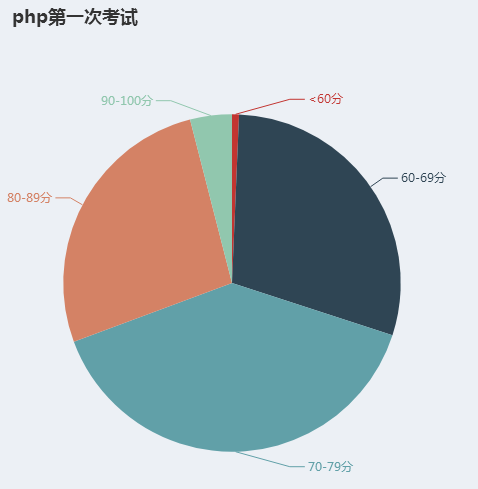
（1）下拉框选择班级，Exam\_manageAction根据exam-manage.jsp传递的班级ID进行查询，且返回处理结果给exam-manage.jsp，exam-manage.jsp显示成员信息。

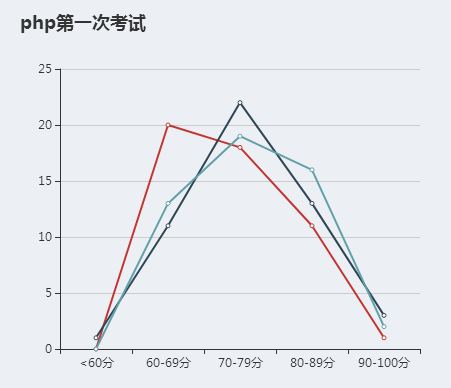
## 教师成绩管理模块

### 考试成绩

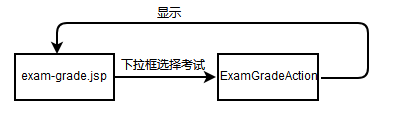
#### 效果图







### 功能流程图



需要说明的问题：

1）饼状图和折线图显示考试成绩信息

### 功能描述

（1）功能类型：查询数据

（2）功能描述：查询考试成绩信息

（3）操作权限：教师

### 界面设计

（1）数据要求

1）功能类型：查询数据

2）数据描述：

页面显示录入字段如下：

下拉框选择考试

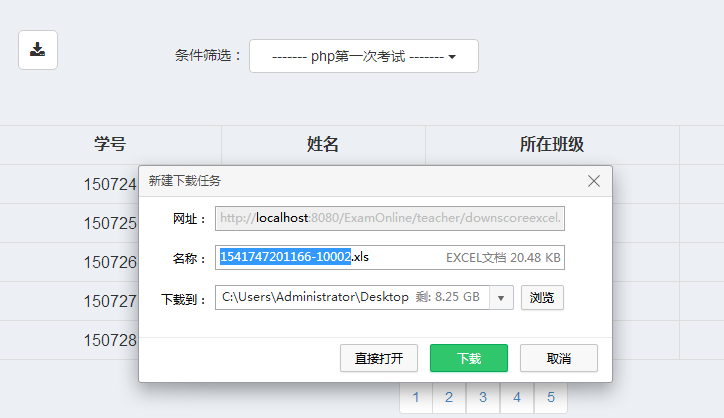
### 内部逻辑实现

选择考试，exam-grade.jsp将考试Id传递给ExamGradeAction，ExamGradeAction进行数据查询并返回处理结果给exam-grade.jsp，subject-library.jsp用饼状图和折线图显示考试成绩信息。

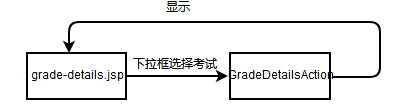
## 成绩明细

#### 效果图

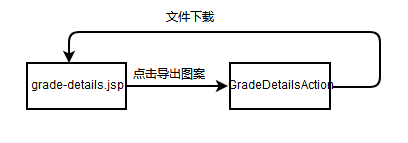




#### 功能流程图



（1）查询



（2）导出成绩

需要说明的问题：

(1)可以删除某次考试某个学生的成绩。

#### 功能描述

（1）功能类型：查询数据，修改数据

（2）功能描述：查询考试成绩信息

（3）操作权限：教师

#### 界面设计

（1）数据要求

1）功能类型：查询数据，删除数据

2）数据描述：

页面显示录入字段如下：

下拉框选择考试

#### 内部逻辑实现

1. 选择考试，grade-details.jsp将考试Id传递给GradeDetailsAction，GradeDetailsAction进行数据查询并返回处理结果给rade-details.jsp，rade-details.jsp显示学生考试成绩信息。
2. 点击删除，grade-details.jsp将学生Id传递给GradeDetailsAction，GradeDetailsAction进行数据处理并返回处理结果给rade-details.jsp，rade-details.jsp显示数据处理结果。
3. 点击导出图案，grade-details.jsp将考试Id传递给GradeDetailsAction，GradeDetailsAction进行数据处理给rade-details.jsp，rade-details.jsp文件下载。

## 管理员学生管理模块

### 效果图

1. 



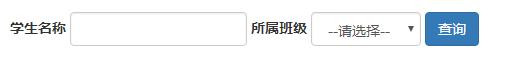




1. 

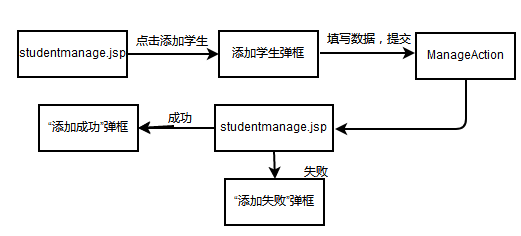


（3）

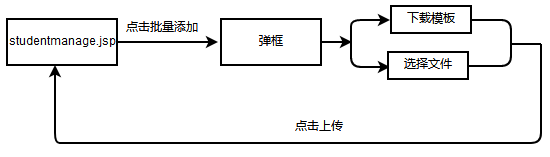


### 功能流程图

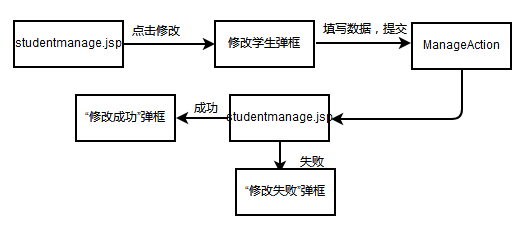
功能流程图如下图所示:



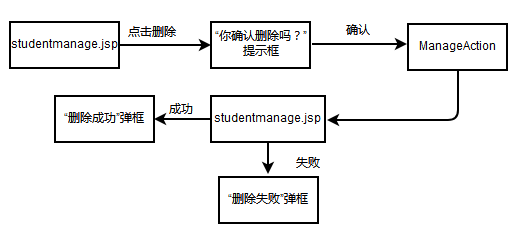
1. 添加学生



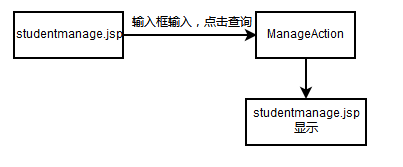
（2）批量添加学生



（3）修改学生



（4）删除学生



（5）查询

（1）需要说明的问题

1）可以对学生姓名、学生班级进行多条件查询。

2）可以对某一学生信息进行更新和删除。

3）点击更新会弹出修改框，修改框内会显示学生数据。

### 功能描述

（1）功能类型：查询数据，更新数据，删除数据，添加数据

（2）功能描述：对学生进行增删查改。

（3）操作权限：管理员

### 界面设计

（1）数据要求

1）功能类型：查询数据，删除数据，更新数据，添加数据

2） 数据描述：

（1）页面显示录入字段如下：

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| studentName | varchar | 学生姓名 |
| studentPwd | varchar | 密码 |
| birthday | date | 出生日期 |
| school | varchar | 所在学校 |
| address | varchar | 家庭住址 |
| phone | varchar | 联系电话 |

显示相应学生信息。

### 内部逻辑实现

1. 打开studentmanage.jsp页面时，ManageAction将全部学生的信息传递给页面，并显示。
2. 页面点击“添加学生”，弹出与之关联的弹框。填写好数据之后，点击“确认增加”，页面利用Ajax将数据交给ManageAction处理，并返回处理结果给页面，界面给出相应的提示框。
3. 页面点击“修改”，传递相应的学生Id给ManageAction，ManageAction获取相应的学生信息给页面，页面加载数据到与之关联的弹框中。编辑好数据之后，点击“确认修改”，页面利用Ajax将数据交给ManageAction处理，并返回处理结果给页面，界面给出相应的提示框。
4. 页面点击“删除”，弹出确认框，再次确认后页面利用Ajax传递相应的学生Id给ManageAction，ManageAction处理后，返回处理结果给页面，界面给出相应的提示框。
5. 在输入框中输入及下拉框选择后，点击查询，页面利用Ajax传递将数据传递给ManageAction，ManageAction处理后将结果返回给页面，页面显示相应数据。

### 存储分配

学生表（studentinfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| studentId | Int 主键 |  |
| studentName | varchar | 学生姓名 |
| studentPwd | varchar | 密码 |
| classId | Int | 班级ID |
| birthday | date | 出生日期 |
| school | varchar | 所在学校 |
| address | varchar | 家庭住址 |
| phone | varchar | 联系电话 |

## 管理员教师管理模块

### 效果图

1. 







（2）



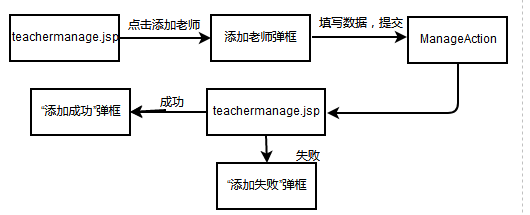


（3）

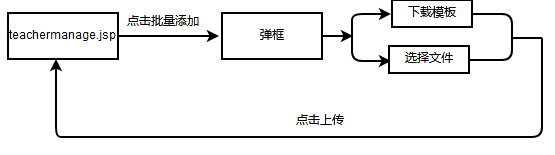


### 功能流程图

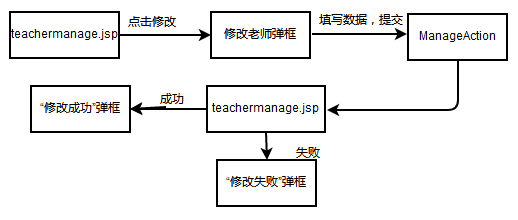
功能流程图如下图所示：



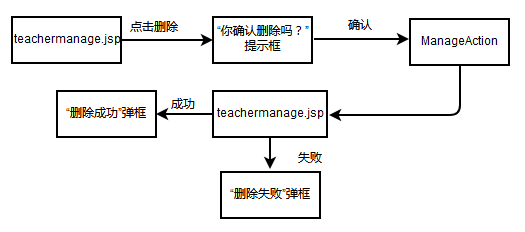
1. 添加老师



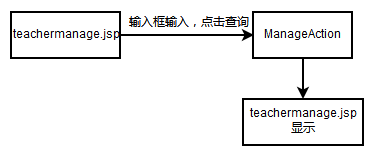
（2）批量添加老师



（3）修改老师



（4）删除教师



（5）查询

1. 需要说明的问题

1）可以对老师姓名进行模糊查询。

2）可以对某一老师信息进行更新和删除。

3）点击更新会弹出修改框，修改框内会显示老师数据。

### 功能描述

* 1. 功能类型：查询数据，更新数据，删除数据，添加数据
  2. 功能描述：对老师进行增删查改
  3. 操作权限：管理员

### 界面设计

* + 1. 数据要求

1）功能类型：查询数据，更新数据，删除数据，添加数据

2）数据描述：

页面显示录入字段如下：

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| teacherName | varchar | 职工姓名 |
| teacherPwd | varchar | 职工密码 |
| joindate | date | 入职日期 |
| email | varchar | 邮箱 |

### 内部逻辑的实现

（1）打开teachermanage.jsp页面时，ManageAction将全部老师的信息传递给页面，并显示。

（2）页面点击“添加老师”，弹出与之关联的弹框。填写好数据之后，点击“确认增加”，页面利用Ajax将数据交给ManageAction处理，并返回处理结果给页面，界面给出相应的提示框。

（3）页面点击“修改”，传递相应的老师Id给ManageAction，ManageAction获取相应的老师信息给页面，页面加载数据到与之关联的弹框中。编辑好数据之后，点击“确认修改”，页面利用Ajax将数据交给ManageAction处理，并返回处理结果给页面，界面给出相应的提示框。

（4）页面点击“删除”，弹出确认框，再次确认后页面利用Ajax传递相应的老师Id给ManageAction，ManageAction处理后，返回处理结果给页面，界面给出相应的提示框。

（5）在输入框中输入，点击查询，页面利用Ajax传递将数据传递给ManageAction，ManageAction处理后将结果返回给页面，页面显示相应数据。

### 存储分配

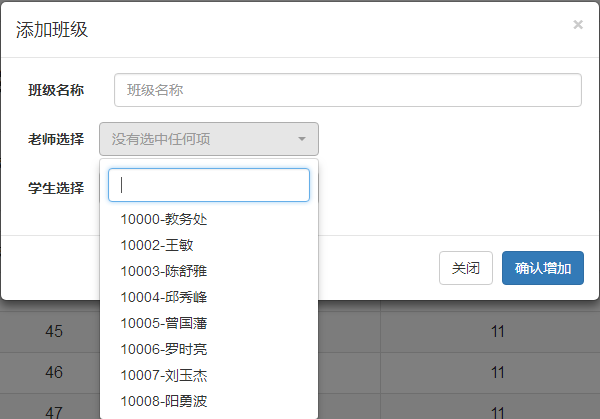
职工表（teacherinfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| teacherId | Int 主键 |  |
| teacherName | varchar | 职工姓名 |
| teacherPwd | varchar | 职工密码 |
| joindate | date | 入职日期 |
| email | varchar | 邮箱 |

## 管理员班级管理模块

### 效果图

1. 



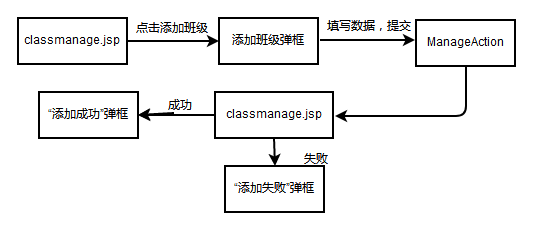
1. 



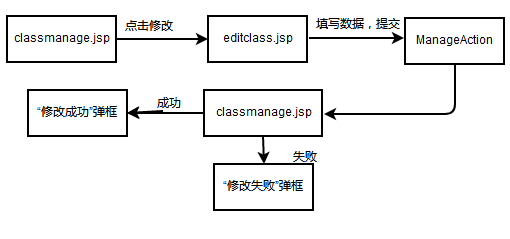
（3）



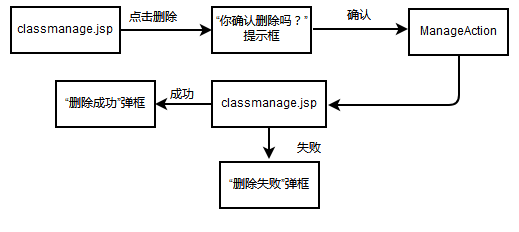
### 功能流程图



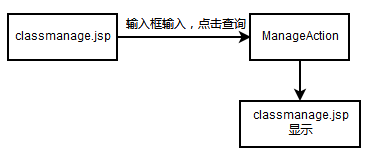
1. 添加班级



1. 修改班级



1. 删除班级



（4）查询

1. 需要说明的问题

1）默认进入页面时显示所有班级。

2）添加班级时需要同时添加班级的教师和学生。

3）点击删除会弹出提示框。

4）修改班级时可以修改班级的所有信息，包括修改选择的老师和学生。

### 功能描述

* 1. 功能类型：查询数据，更新数据，删除数据，添加数据
  2. 功能描述：对班级进行增删查改
  3. 操作权限：管理员

### 界面设计

* + 1. 数据要求

1）功能类型：查询数据，更新数据，删除数据，添加数据

2）数据描述：

页面显示录入字段如下：

（1）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| className | varchar | 班级名称 |

（2）下拉框多选学生和老师。

### 内部逻辑的实现

（1）打开classmanage.jsp页面时，ManageAction将全部老师的信息传递给页面，并显示。

（2）classmanage.jsp页面点击“添加班级”，弹出与之关联的弹框。填写好数据之后，点击“确认增加”，页面利用Ajax将数据交给ManageAction处理，并返回处理结果给页面，界面给出相应的提示框。

（3）页面点击“修改”，跳转到editclass.jsp传递相应的老师Id给ManageAction，ManageAction获取相应的老师信息给页面，页面加载数据到与之关联的弹框中。编辑好数据之后，点击“确认修改”，页面利用Ajax将数据交给ManageAction处理，并返回处理结果给页面，界面给出相应的提示框,然后跳转到classmanage.jsp。

（4）classmanage.jsp页面点击“删除”，弹出确认框，再次确认后页面利用Ajax传递相应的老师Id给ManageAction，ManageAction处理后，返回处理结果给页面，界面给出相应的提示框。

（5）在输入框中输入，点击查询，页面利用Ajax传递将数据传递给ManageAction，ManageAction处理后将结果返回给页面，页面显示相应数据。

### 存储分配

（1）

班级表（classinfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| classId | Int 主键 |  |
| className | varchar | 班级名称 |

（2）

学生表（studentinfo）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| studentId | Int 主键 |  |
| studentName | varchar | 学生姓名 |
| studentPwd | varchar | 密码 |
| classId | Int | 班级ID |
| birthday | date | 出生日期 |
| school | varchar | 所在学校 |
| address | varchar | 家庭住址 |
| phone | varchar | 联系电话 |

（3）

职工班级联系表（classandteacher）

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 解释 |
| ctid | Int 主键 |  |
| teacherId | int | 职工编号 |
| classId | int | 班级编号 |