



江西师范大学

JIANGXI NORMAL UNIVERSITY

学士学位论文

THESIS OF BACHELOR

(2011—2015 年)

基于 JSP 的在线考试系统的设计与实现  
——学生功能模块

Design and Implementation of Online Test System

Based on JSP

-----Student Module

指导老师： 刘 媛 春

姓 名： 熊 心

学 号： 1167003073

学 院： 软件学院

专 业： 软件工程

完成时间： 2015 年 4 月

## 摘 要

学校对学生的考核是教学管理的一个重要内容，而大量印刷式的纸质考试很容易造成资源的浪费。现在，传统的管理方法、手段以及工作效率已不能适应新的发展需要，形式单调，学生积极性不高。在线考试系统的应用提高了学校考试的丰富性和管理的效率。

此在线考试系统是基于 MVC 架构，使用 JSP 实现的系统，后台数据库采用的是 MySQL。系统有两个用户即管理员和学生，学生用户可以查看考试安排，选择进行考试，查看考试成绩，修改用户密码。管理员的功能有题目管理、试题管理、用户管理和成绩管理。本文首先简单介绍了系统的开发背景，然后详细阐述了系统的设计与实现过程，着重介绍了个人模块即学生功能模块。

**关键词：** 在线考试；MVC；JSP；MySQL；学生功能

# Abstract

School student assessment is an important part of teaching management, and the paper printing method examination is easy to waste resources. The traditional management methods and efficiency can not satisfy the needs of the new development in nowadays. Because of monotonous, students are not enthusiastic. With the Applying of the Online Test System have improved the variety of school examination and management efficiency.

This Online Test System is based on MVC architecture and uses JSP and MySQL database to realize. The system has two users whose are administrators and students. Students can search the exam schedule, selects to participate the test, watches the results ,updates the passwords. The administrators part have a problem management module, test management module, user management module and performance management modules, in addition to the basic add, delete, change, but also has more detailed results ranking. This paper briefly introduces the development background of the system, the techniques used and elaborated system design and implementation process, intensify to introduce the students module.

**Key words:** Online Exam; MVC; JSP; MySQL ; Student Module

# 目 录

第 1 章 绪论.....	1
1.1 系统的开发背景.....	1
1.2 系统的开发目标.....	1
1.3 本文的主要内容.....	2
第 2 章 系统的分析与设计.....	3
2.1 系统数据流图.....	3
2.2 系统功能模块图.....	4
2.3 个人模块的设计.....	4
2.4 数据库设计.....	10
2.4.1 E-R 图.....	10
2.4.2 数据表.....	11
第 3 章 系统个人模块的实现.....	13
3.1 主页的实现.....	13
3.2 登录模块的实现.....	15
3.3 注册模块的实现.....	17
3.4 考试模块的实现.....	20
3.5 密码管理模块的实现.....	24
3.6 分数查看的实现.....	26
第 4 章 总结.....	29
参考文献.....	30

# 第1章 绪论

## 1.1 系统的开发背景

百年大计，教育为本。教育是立国之本，民族兴旺的标记，一个国家有没有发展潜力看的是教育，这个国家富不富强看的也是教育。而考试是教育中的一个重要环节，是一种严格的知识水平鉴定方法。要求考生在规定的时间内按指定的方式解答精心选定的题目或按主办方的要求完成一定的实际操作的任务，并由主办方评定其结果，从而为主办方提供考生某方面的知识或技能状况的信息。

考试的目的一是为了考察学生的知识掌握情况，二是为教师提供教学分析的依据。传统的考试由于涉及到组织命题、试卷应刷、考场安排、组织阅卷等诸多环节，考试时间周期长、效率低下。同时人工阅卷等主观原因也影响到考试的公正性。随着网络技术在教育领域运用的普及，应用现代信息技术构架的网络在线考试系统展现除了越来越多的优越性。

网络在线考试系统相对于传统的考试方式，具有自动控制考试时间、系统自动评分、有效防止舞弊等优点：实现了考试功能与教育评估和教育力能力的高度结合；减轻了教务人员的工作负担，提高了工作效率；使考务工作自动化，保证出卷、考试和考生管理各阶段正常有序的进行，使考试更趋于客观、合理和公正。另外无纸化办公的形式，降低了考试成本，低碳环保。使用 B/S 结构，可以跨平台，易于维护。

## 1.2 系统的开发目标

JSP 在线考试系统开发基本目标是将后台中管理员自己出的题库正确、迅速的显示在前端页面，供用户考试，用户可以自己选择想要考试的学科进行考试，并得出成绩，让用户对自己学习的成果有一个大概的了解。前端页面通过与后台数据交互为用户提供注册登录等功能，并在主页有一个专门的查看分数的栏目以及显示当前正在进行中考试科目的栏目，从而能够更顺利地进行考试。

为了有效的发挥在线考试系统的作用，有效的帮助人们解决在考试的时候遇到的各种问题。功能上对在线考试系统界面的要求是全面的，可以分为以下三点：

（1）让用户可以使用手机，平板，网页，等使用我们的考试系统，时刻方便用户使用。（2）用户可以在上面考任何知识，可以是任何专业知识。（3）界面简洁友好。

在 JSP 在线考试系统界面设计和开发过程中，要注意系统界面的安全性，界面处理的准确性和及时性。充分考虑系统当前和将来可能承受的工作量，使界面的处理能力和响应时间能够满足对信息处理的需求。

### 1.3 本文的主要内容

考虑到纸质试卷易丢失，不环保，浪费大量人力等缺点，所以开发一个在线考试系统供学生考试。由于 Java 在 windows 和 linux 都能安装，而且用 JSP 有强大的 J2EE 平台，方便的 JDBC 接口。在 windows 和 linux 系统都能用 MyEclipse 来进行编码和调试 Java 和 JSP 程序。所以本系统应用了 JSP 技术，MVC 框架，tomcat+MYSQL 服务器部署，用的 MyEclipse 开发和调试。

本在线考试系统的用户分为管理员和学生。学生用户有查看考试安排，选择进行考试，查看考试成绩，修改用户密码等功能。管理员用户有对题库进行增删改查，对考试结果进行管理，注册用户信息管理，当前管理员用户的密码修改等功能。

本系统题库由管理员提供，亲自出题，同时避免学生在网上查找答案，影响考试的公正公平性。并且本考试系统能够自动评卷，试题种类丰富，不受学科领域限制，无需人工干预，全自动化，能够快速审阅试题，提高了办公效率，能够实时的计算成绩并针对当前用户给出成绩排行。

第 1 章，简明的介绍系统的开发背景、系统的开发目标，并分析了开发该系统应进行的工作。

第 2 章，分析了 JSP 在线考试系统的应用需求，得出系统数据流图、系统功能结构图、系统用例图，并进行了数据库的设计。

第 3 章，详细的介绍了主页、登录模块、注册模块、考试模块、密码管理模块、分数查看功能的实现。

第 4 章，对系统进行了总结并给出了下一步工作。

## 第2章 系统的分析与设计

### 2.1 系统数据流图

数据流图是结构化分析的基本工具。一个数据流图可以标志系统的转换过程，系统所操纵的数据或物质集合，以及过程，存储和外部世界之间的数据流或者物质流。通过对前端页面功能的详细分析，得到的数据流图。

在线考试系统的用户分为三类，学生用户、游客和管理员用户。管理员可以管理题目，也可以管理试题，可以管理考试结果，管理注册的用户。用户可以进行注册，可以选择试题进行考试，完成考试后系统能自动评阅，用户和管理员都可以查阅考试成绩。系统数据流图如图 2.1 所示。

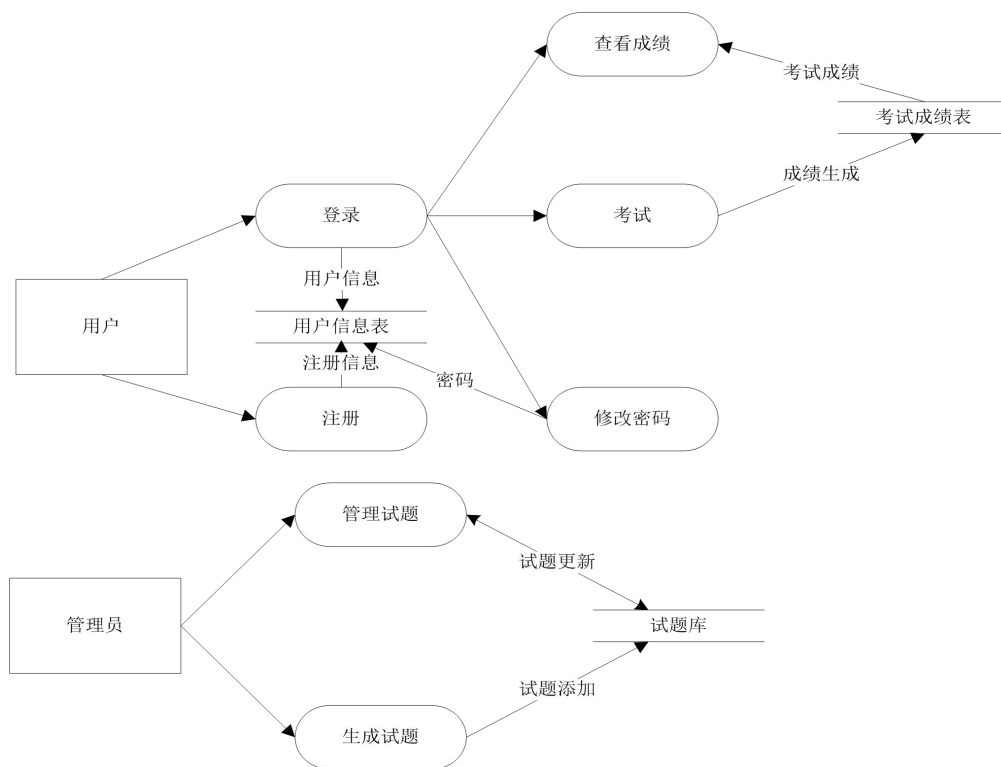


图 2.1 系统数据流图

## 2.2 系统功能模块图

系统主要分为管理员和学生用户，学生用户可以注册新账户，选择考试试题进行考试，查询考试成绩，修改账户密码；管理员可以对题库进行添加和删除，对试题进行查找、添加、删除和预览，对考试结果进行查询，对已注册的用户进行信息查找和删除。系统总体功能结构如图 2.2 所示：

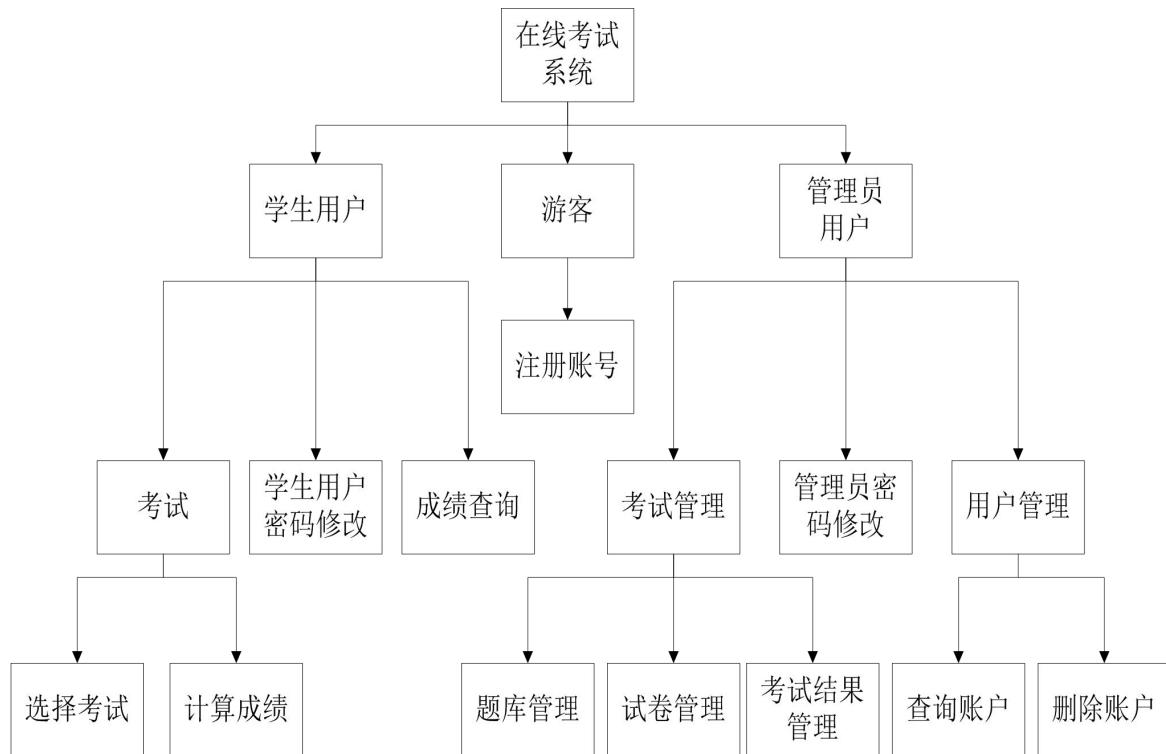


图 2.2 系统功能结构图

## 2.3 个人模块的设计

### （1）功能模块图

本人负责的主要为用户模块，主要包括主页模块、登录模块、注册模块、考试模块、密码管理模块、成绩查看模块的分析设计和实现。其功能模块图如图 2.3 所示。



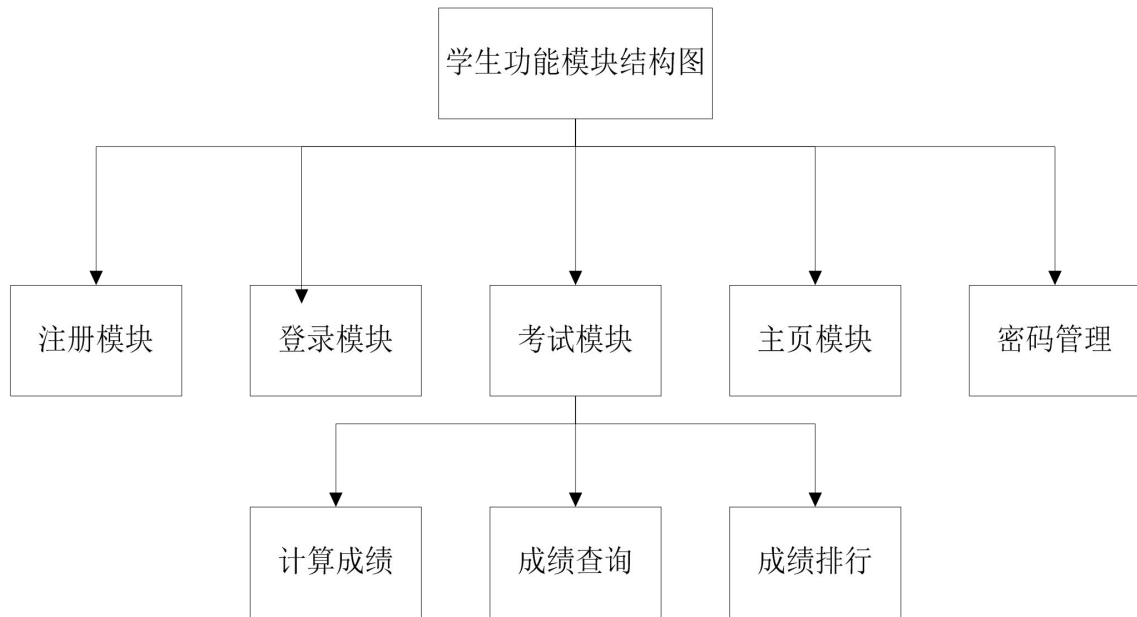


图 2.3 学生功能模块图

### A、登录模块

用户进入登录页面后，可以选择登录账号进入主页。登录时需要输入用户账号以及密码，当登录账号或者密码输入错误时，会报错，弹出用户或密码错误信息，以及一个确定按钮，点击确定按钮，可以重新回到登录模块。登录模块还提供了重置功能，可以把当前输入框中的用户账号以及密码清空，不需要繁琐的一个个删除，节省用户的时间，提高用户体验。

### B、考试模块

当用户登录进入后，点击考试栏目，考试界面会以表格形式列出，系统根据考试时间列出所有存在的考试项，当光标经过当前时间在考试时间之内的可以考试的考试项目时，可以点击试题名称进行考试，否则只能查看考试名称、开始时间和结束时间，用户可以选择自己想要进行的考试科目进行考试，只要考试科目满足考试条件。考虑到自动评卷和免人工批改的问题，可以限定只添加选择题作为考核，后续会添加相应的主观题，这样可以直接计算成绩并显示该用户成绩排行。用户进入考试科目后，在认真的完成客观选择题后，经过反复检查点击提交按钮，即完成了试卷提交。系统会立即计算该试卷的成绩，成绩的计算方法为对的客观题数除以题目总数然后乘以 100，成绩会精确到小数点后 4 位，保证了成绩的精准性。

### C、注册模块

用户进入登录页面后，可以选择注册账号。注册时需要填写用户的相关信息包括：用户名，姓名，密码，重复密码，性别，电话，邮箱地址，备注。用户名唯一标识一个用户账号，用户名，姓名，密码，重复密码以及性别是必填的，电

话，邮箱地址以及备注是可选的，当用户不填时，注册模块正常运行。当用户名不填时，点击提交按钮会出现“用户名不能为空”的提示信息，点击确定按钮重新回到注册页面，当其他必填的信息不填时，也会出现类似的提示信息，这里不再重复介绍。同时为保证密码的有效性，必须密码和重复密码栏保持一致，当两次密码输入不一致时，会弹框提示“两次密码不一致”的错误信息，另外也提供了重置按钮可以清空输入框，提高了用户体验，在注册界面也可以不注册直接有链接跳转至登录界面。在正确填写完信息之后点击提交之后会有注册成功的弹窗提示，之后就自动跳转至登录界面进行登录。

### D、密码管理模块

当用户登录系统之后，进入主页后可以点击修改密码菜单键进行修改，当然是要在登录未过期的情况下才能修改。修改界面将自动会列出当前要修改的用户名，不需要用户手动输入用户名，更具人性化，提高了用户体验，让用户能更好的使用本系统，当两次输入的密码相同时才会保存新密码并提示修改成功，修改成功后会将最新密码录入数据库，否则修改失败并提示两次密码输入不一致，并需要重新修改密码。

### E、主页模块

用户登录系统之后，直接进入主界面，主界面分为三个区域，上方显示系统名称和 logo 等信息；左边显示功能列表，以表格形式列出用户所能进行的操作，分别为考试，分数查看，密码修改以及退出栏目；右边为显示的主体区域，都是响应左边菜单点击事件的显示区域。这样可以隐藏项目的结构，给用户提供一个统一的界面，呈现网站的整体风格。当用户点击考试栏目时，会在右边的主体区域显示当前存在的考试科目栏，即没有被管理员删除的考试科目，考试科目栏包括考试科目的名称以及考试科目的开始时间和结束时间，当用户进入考试科目的时间大于该考试科目的开始时间以及小于结束时间时，即满足考试条件，就可以进行考试，否则不能进行考试，只能选择进入其他符合条件的考试。当用户点击分数查看栏目时，会在右边的主体区域显示用户所有进行过的考试成绩，成绩这一栏会显示考试科目的名称，该考试科目的开始时间，以及用户的成绩，成绩栏会按成绩的高低先后顺序排列，最高的成绩显示在第一行，然后逐层递减，最低的成绩显示在最后一行。当用户点击密码修改栏目时，即可进行密码修改，前面已经详细介绍了，所以这里不再重复介绍。最后，当用户点击退出栏目时，页面会直接退出主页栏目，直接进入登录模块。

### (2) 用例图

用例图提供了对用户需求的高级可视化表示，一幅用例图包含的元素有：系统、行为者、用例以及用例之间的关系。在经过对系统功能的详细分析后，得到用户和游客用例图如图 2.4 所示。

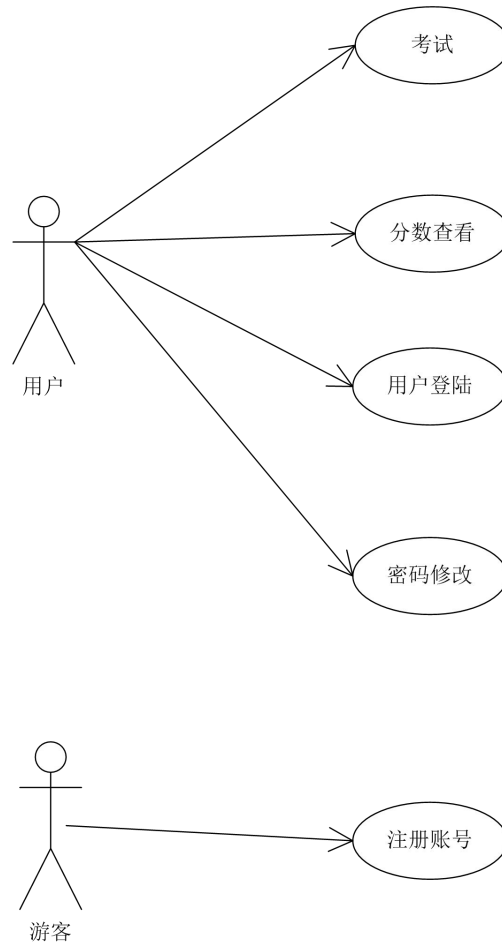


图 2.4 用户和游客用例图

用户使用前台界面进行考试，前提是在登录考试系统的情况下，点击考试菜单后，选择考试列表中的考试项目，进行考试，完成考试可以是用户主动提交，也可以是考试时间到，系统自动提交，用户考试用例描述如表 2.1 所示。

表 2.1 用户考试用例表

描述项	说明
用例名称	考试
用例描述	描述了用户使用本系统前台子系统试卷选择及进行考试的整个过程
参与者表	客户（用户）

状态*	进行中
前置条件	用户已登录系统
后置条件	考试时间在范围内
基本操作流	1.用户在系统主页上选择试卷分类，进入试卷列表查看界面或在搜索框中选择要购进行的考试名称进行检索，检索得到符合条件的试卷列表； 2.找到想进行的考试后点击“查看试卷详细信息”按钮，进入试卷详细信息的查看页面； 3.确定进行考试后，等到考试时间到达，点击“开始考试”按钮，进行考试；
可选操作流	1. 用户将所选择的答案填完后，若不想继续进行考试，则可移动到页面下方，确认无误后，点击提交试卷按钮，进入计算成绩界面。 2. 用户将所参加的考试试卷提交后，可以立即查看所获得的成绩。
被泛化用例表	无
被包含用例表	用户登录
被扩展用例表	分数查看、考试结果查询

用户登录系统后，用户可以选择任意一场已参加过的考试进行查看，选择相应的试卷，就可以查看考试的详细信息，或是用户完成考试后，系统自动跳转至分数查看界面列出成绩排行。用户的分数查看用例描述如表 2.2 所示。

表 2.2 分数查看用例表

描述项	说明
用例名称	分数查看
用例描述	描述了用户查看自己考试成绩及其详细信息的过程

参与者表	客户（用户）
状态*	进行中
前置条件	用户已登录系统
后置条件	结束考试，并交卷
基本操作流	1.用户在系统主页上选择查看成绩，进入试卷列表查看所参加考试的成绩。 2.选择相应的试卷，进入查看考试的详细信息。
可选操作流	用户可以选择任意一场已经参加过的考试进行查看
被泛化用例表	无
被包含用例表	用户考试、用户登录
被扩展用例表	考试结果查询

游客只能看到登录注册界面，注册账号是要输入必要的用户名、密码、性别、邮箱等，而电话号码和备注则为选填信息。游客的非用户注册用例描述如表 2.3 所示。

表 2.3 非用户注册用例表

描述项	说明
用例名称	非用户注册
用例描述	描述了非用户对该系统进行注册的过程
参与者表	游客
状态*	进行中
前置条件	非用户登录网站
后置条件	系统提示操作成功
基本操作流	非用户在文本框中输入用户名 非用户在文本框中输入密码并确认 非用户在文本框中输入邮箱

	非用户在选择框中选择性别
可选操作流	非用户在文本框中输入中文名 非用户在文本框中输入电话号码 非用户在文本框中输入备注信息
被泛化用例表	无
被包含用例表	无
被扩展用例表	用户登录、考试、查看成绩

## 2.4 数据库设计

### 2.4.1 E-R 图

在线考试系统中的实体联系如下：用户选择试题考试才能产生考试成绩，一个管理员可以查看多个用户的成绩和管理多个注册的账户，一个管理员可以管理多个试题，而一份试题由一到多道题目构成，用户可以查询多个考试成绩，用户也可能在同一份试题上做到相同的考试分数。所以得到系统实体-联系图如图 2.5 所示。

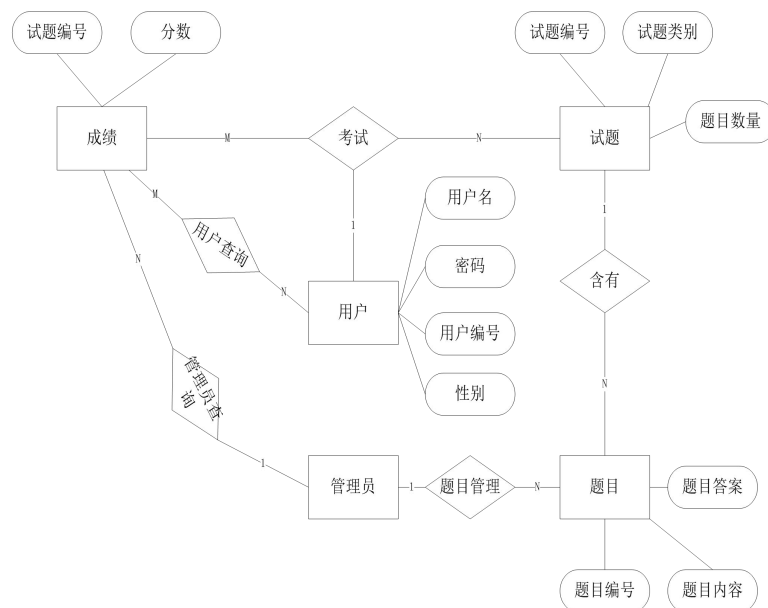


图 2.5 系统实体-联系图

### 2.4.2 数据表

根据实体-联系图得到数据库模式设计如下。

试卷（试卷编号，试卷类别，题目数量，分数）。

用户（用户名，密码，用户类型，性别，电子邮件，电话号码）。

题目（题目编号，题目内容，题目类别）。

查询（用户名，试卷编号，分数）。

试卷与题目（试卷编号，题目编号）。

从而得到各数据表的设计如下。

（1）用户信息表（USER\_INFO）：每一个账号都有唯一的编号，所以设置id为主键，用户名和密码都设置为可变字符类型，如表2.4所示。

表 2.4 用户信息表

字段名称	数据类型	主键	描述
id	int	是	用户编号
username	varchar(16)	否	用户名
pwd	varchar(16)	否	密码
name	varchar(16)	否	用户姓名
sex	char(2)	否	性别
telephone	varchar(50)	否	电话号码
email	varchar(100)	否	邮箱地址
remark	text	否	备注

（2）题目信息表（QUESTIONS）：每道题目都有唯一的id号，且题目内容和答案都为可变字符且不能为空，题目信息表如表2.5所示。

表 2.5 题目信息表

字段名称	数据类型	主键	描述
id	int	是	题目编号
qName	varchar(100)	否	题目内容
qAnswer	char(1)	否	题目答案

（3）题目选项表（OPTIONS）：题目的每个选项都是不能为空的，且每个选项要有对应的选项编号或选项名称，每个题目选项都对应一道题目，所以需要给qId添加外键约束。题目选项表如表2.6所示。

表 2.6 题目选项表

字段名称	数据类型	主键	描述
id	int	是	选项表编号
oNO	char(1)	否	选项编号
oName	varchar(100)	否	选项内容
qId	int	否，外键	题目编号

(4) 考试名称表 (TEST\_PAPER)：每一场考试都有唯一的考试编号，有考试名称、开考时间和交卷时间。所以考试编号要有主键约束。考试名称表如表 2.7 所示。

表 2.7 考试名称表

字段名称	数据类型	主键	描述
id	int	是	考试编号
tName	varchar(100)	否	考试名称
startTime	datetime	否	开考时间
endTime	datetime	否	交卷时间

(5) 考卷信息表 (TEST\_QUESTIONS)：一份考卷的信息是由多道考题组成的，且每份考卷都有一份考卷名称与其对应，所以 tId 和 qId 既是主键也是外键。考卷信息表如表 2.8 所示。

表 2.8 考卷信息表

字段名称	数据类型	主键	描述
tId	int	是，外键	题目编号
qId	int	是，外键	考试名称编号

(6) 考试成绩表 (TEST\_RESULT)：每份考试成绩表都有自己的编号，还要有考试的名称信息，和参加考试的用户的编号，所以都应该有对应的外键约束。并有考试成绩记录。考试成绩表如表 2.9 所示。

表 2.9 考试成绩表

字段名称	数据类型	主键	描述
id	int	是	考试成绩表编号
tId	int	否，外键	试卷编号



userId	int	否，外键	用户编号
mark	float	否	考试成绩

## 第3章 系统个人模块的实现

### 3.1 主页的实现

在任何 WEB 站点上,主页是最重要的页面,会有比其他页面更大的访问量。有很多形象的比喻可以说明主页的作用:主页是杂志的封面;主页是对外的脸面;主页是一种艺术;主页就像是门厅;主页就像书的目录;主页就像一本小册子。所有上述的比喻都在一定程度上反映了主页的特点,但每个比喻事物都与主页有本质上的不同。主页的目的是多样的,访问者的目的也是多样的。我们的设计重点突出了一目了然,又要充分理解访问者的目的,这都是设计主页的关键。

功能特色及实现技术:

界面简洁美观,另外随着广告网页和网站越来越多,很多网页网站都或多或少的带了一些广告宣传,给用户带来了诸多不便,对学习性网站影响更加严重,影响网站的好评率,而该主页直入主题,直接显示学习的相关知识,让用户更快的进入学习状态。使用 frameset 框架集,把页面划分成上下两部分,在下面部分的区域内再嵌套一个框架,分为左右两部分,完成了布局。左边部分用来存放功能菜单,右边部分为响应菜单功能的主体区域。主页界面如图 3.1 所示。



图 3.1 主页界面

主页设计的核心代码如下:

```
<frameset rows="10%,*" cols="*" frameborder="no" border="0" framespacing="0">
  <frame src="../top.html" name="topFrame" scrolling="No" noresize="noresize"
id="topFrame" title="topFrame" />
  <frameset cols="10%,*" frameborder="no" border="0" framespacing="0">
    <frame src="left.jsp" name="leftFrame" scrolling="No" noresize="noresize"
id="leftFrame" title="leftFrame" />
```

```
<frame src="testAction.jsp?action=query" name="mainFrame" id="mainFrame"
title="mainFrame" />
</frameset>
</frameset>
```

## 3.2 登录模块的实现

当用户输入网站访问到登录界面时，有两个输入框三个按钮，当用户输入的账号不存在则无法登录，可以点击注册按钮跳转至注册界面输入基本信息点击提交后将保存信息至用户信息表并返回登录页面。当输入账号或密码为空时，系统将提示请输入用户名密码，输入错误的账号密码信息时将显示用户密码错误，直至账号密码匹配才能正确的登录系统。用户登录模块的流程图如图 3.2 所示。

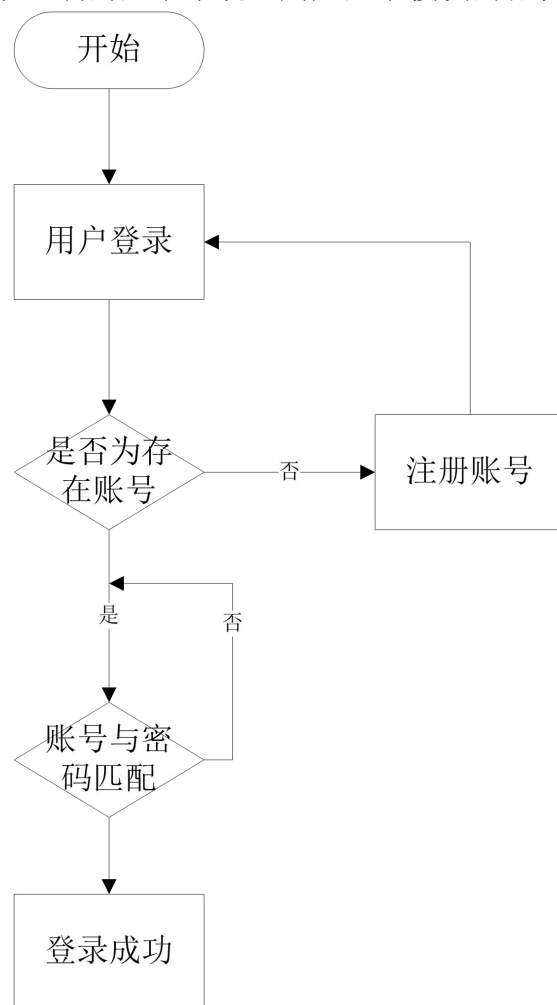


图 3.2 用户登录模块流程图

实现的登录界面如图 3.3 所示。

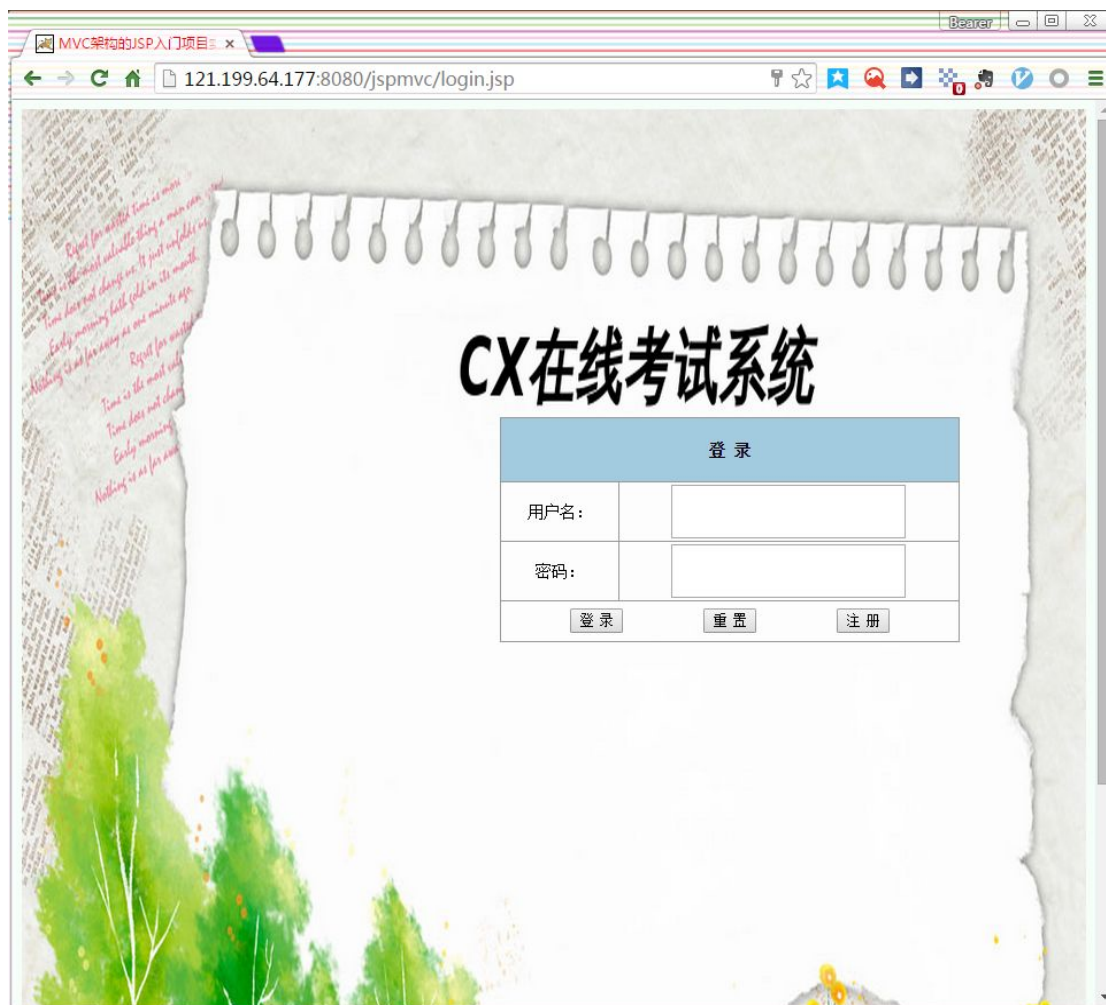


图 3.3 登录界面

考虑到登录系统的安全，防止 sql 注入，采用了占位符?设参数的方式查询。用户登录模块实现的核心代码如下：

```
public UserInfo login(String username ,String pwd){  
    Connection conn = null;  
    PreparedStatement pst = null;  
    try{  
        //获取连接  
        conn = DBManager.getConnection();  
        //sql 语句  
        /*这里最好使用占位符?设参数的方式查询，避免 sql 注入
```

```
String sql = "select * from USER_INFO where username = ? and  
pwd= ?";  
  
pst = conn.prepareStatement(sql);  
pst.setString(1, username);  
pst.setString(2, pwd);  
//设置参数  
ResultSet rs = pst.executeQuery();  
if(rs.next()){  
    UserInfo userInfo = new UserInfo();  
    userInfo.setId(rs.getInt("id"));  
    userInfo.setUsername(rs.getString("username"));  
    userInfo.setPwd(rs.getString("pwd"));  
    userInfo.setName(rs.getString("name"));  
    userInfo.setSex(rs.getString("sex"));  
    userInfo.setEmail(rs.getString("email"));  
    userInfo.setTelephone(rs.getString("telephone"));  
    userInfo.setRemark(rs.getString("remark"));  
  
    return userInfo;  
}  
} catch (Exception e) {  
    e.printStackTrace();  
}
```

### 3.3 注册模块的实现

在用户注册界面输入对应信息，输入完成点击注册后，通过 form 表单调用 function 方法验证输入信息是否符合输入规范，验证不符合则提示相应输入错误规范，停在原界面；若验证符合输入规范，则通过 form 表单和相应 jsp 文件接收输入的数据，然后在 action 中调用 dao 方法验证输入的名称与密码是否和数据库表中相对应，通过注册则提示注册成功，否则提示输入的错误信息，继续停留在

登录界面。用户注册程序流程图如图 3.4 所示：

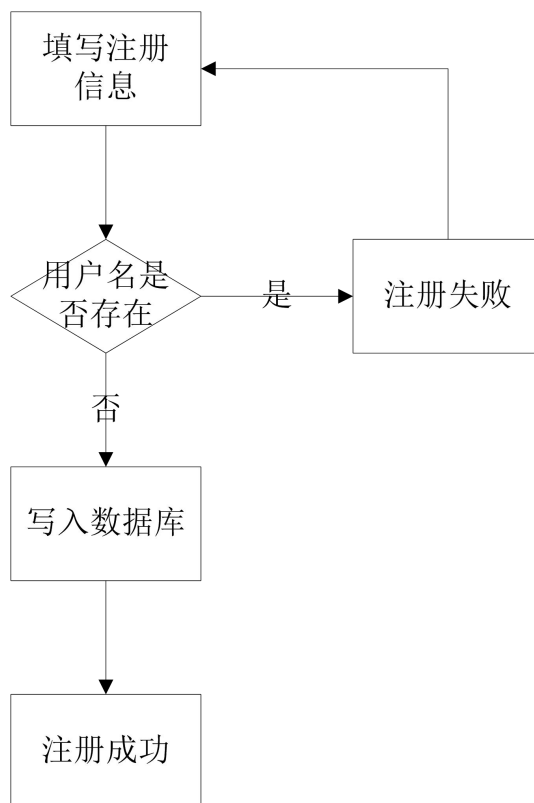


图 3.4 注册流程图

注册界面如图 3.5 所示。

用户注册			
*用户名:	<input type="text"/>	*姓名:	<input type="text"/>
*密码:	<input type="password"/>	*重复密码:	<input type="password"/>
*性别:	<input type="text" value="男"/>	电话:	<input type="text"/>
*邮箱地址:	<input type="text"/>		
备注:	<input type="text"/>		
<input type="button" value="提交"/> <input type="button" value="重置"/> <input type="button" value="登录"/>			

图 3.5 用户注册界面

用户注册实现部分代码如下：

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    // TODO Auto-generated method stub
```

request.setCharacterEncoding("UTF-8");//统一设置编码，不用再重复写转编码语句

```
String username = request.getParameter("username");
String pwd = request.getParameter("pwd");
MD5 md5 = new MD5();
pwd = md5.getMD5ofStr(pwd);
String name = request.getParameter("name");
String sex = request.getParameter("sex");//性别 男女
String telephone = request.getParameter("telephone");//电话
String email = request.getParameter("email");//邮件
String remark = request.getParameter("remark");//备注
//封装信息到 userInfo
UserInfo userInfo = new UserInfo();
userInfo.setUsername(username);//设置用户名
userInfo.setPwd(pwd);
userInfo.setName(name);
userInfo.setSex(sex);
userInfo.setTelephone(telephone);
userInfo.setEmail(email);
userInfo.setRemark(remark);//计算成绩
UserInfoDao userInfoDao = new UserInfoDao();
if(userInfoDao.isExistUsername(username, 0)){
    request.setAttribute("error", "注册失败:用户已存在");
    request.getRequestDispatcher("/register.jsp").forward(request, response);
}else{
    try {
        userInfoDao.add(userInfo);
        response.setContentType("text/html; charset=UTF-8");//统一编码
        PrintWriter out = response.getWriter();
```

```
out.println("<html><head></head><body>");
out.print("<script type=\"text/javascript\" language=\"javascript\">");
out.print("alert('注册成功');");
out.print("window.location='login.jsp';");//跳转
out.print("</script>");
out.print("</body></html>");
} catch (Exception e) {
// TODO Auto-generated catch block
e.printStackTrace();
request.setAttribute("error", "保存信息失败:"+e.getMessage());
request.getRequestDispatcher("/register.jsp").forward(request,response);
}
}
}
```

### 3.4 考试模块的实现

先获取当前用户的用户名，如果点击考试，则连接数据库查询并以列表方式列出是否有考试安排，且用 forward 标签来跳转至考试安排列表页面。选择有效的考试项跳转到 testPaper.jsp 进行考试，点击提交按钮后获取当前页面提交的答案，并封装为试卷便计算出分数后保存再主动跳转到成绩列表界面显示成绩。分析得到的考试流程图如图 3.6 所示。



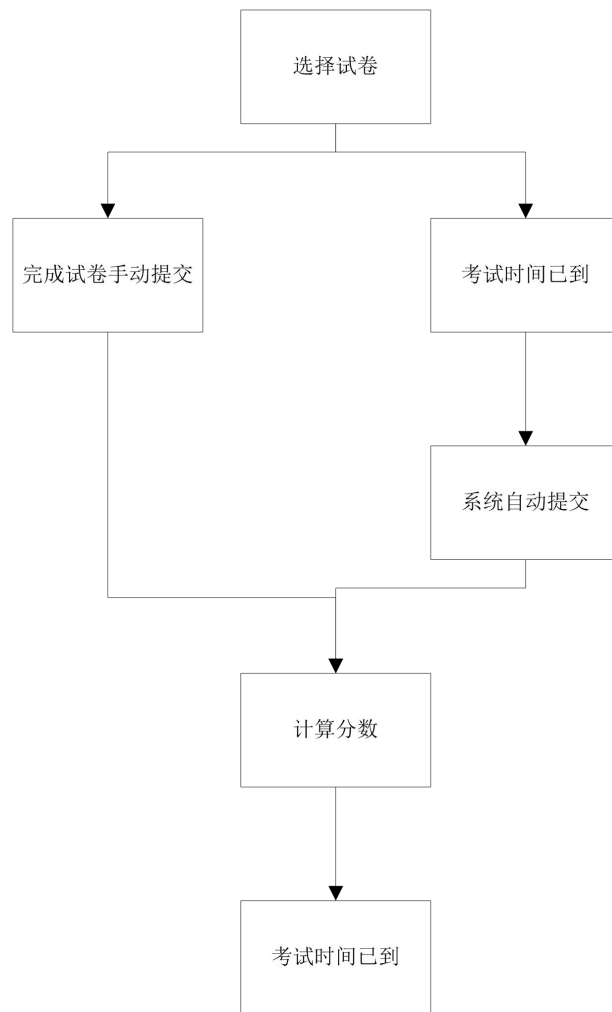


图 3.6 考试流程图

考试界面如图 3.7 所示。

CX在线考试系统		
math (考试时间: 12:43 ~12:41)		
<b>功能列表</b> 考试 分数查看 密码修改 退出	1、test1	
	<input type="radio"/> a、aaaaaaaaa	
	<input type="radio"/> b、bbbbbbbbbbb	
	<input type="radio"/> c、ccccccccccc	
	<input type="radio"/> d、ddddddddddd	
<input type="radio"/> e、eeeeeeeeeee		
2、test2		
<input type="radio"/> a、aaaaaaa		
<input type="radio"/> b、bbbbbbbbbbbbbb		
<input type="radio"/> c、cccccccccccccccccccc		
3、test3		
<input type="radio"/> a、slfgkjslk		
<input type="radio"/> b、aslgjslkgj		
<input type="radio"/> c、slgjsdlgljsdk		
4、test4		
<input type="radio"/> a、5		
<input type="radio"/> b、4		
<input type="radio"/> c、3		
<input type="radio"/> d、2		
<input type="radio"/> e、1		
<input type="button" value="提交试卷"/>		

图 3.7 考试界面

核心代码如下：

```
request.setCharacterEncoding("UTF-8");
UserInfo user = (UserInfo)session.getAttribute("user");//取得用户
String action = request.getParameter("action");
TestPaperDao tpDao = new TestPaperDao();
if (action == null||"query".equals(action)) { //查询
    List list = tpDao.showPaperList(user.getId());
    request.setAttribute("list",list);
<!-- 采用标签 forward -->
    <jsp:forward page="test_list_view.jsp"></jsp:forward>
}
else if("doTest".equals(action)){
    int id = 0;
    try{
        id = Integer.parseInt(request.getParameter("id"));
    } catch(Exception e){
        out.println(e);
    }
    TestPaper testPaper = tpDao.getById(id);
    request.setAttribute("testPaper", testPaper);
request.getRequestDispatcher("/inc/testPaper.jsp").forward(request, response);
}
else if("submitTest".equals(action)){
    TestPaper testPaper = new TestPaper();
    int id = Integer.parseInt(request.getParameter("id"));
    testPaper.setId(id);
numeration et = request.getParameterNames();//取得所有提交的 parameter 参数
while(et.hasMoreElements()){ //取得下一个参数
    String pName = (String)et.nextElement();//参数名
```

```
        if(pName.startsWith("qId")){
            String value = request.getParameter(pName);
            int qId = Integer.parseInt(pName.substring(3));
            //构建，获取提交的答案,并封装为试卷
            Questions questions = new Questions();
            questions.setId(qId);
            questions.setqAnswer(value);
            testPaper.getQuestionsList().add(questions);
        }
    }

    TestPaperService ts = new TestPaperService();
    float mark = ts.mark(testPaper);//得到分数
    TestResult testResult = new TestResult();
    testResult.setTestPaper(new TestPaper(id));//试卷 id
    testResult.setUserInfo(user);//用户 id
    testResult.setMark(mark);//分数
    //保存
    new TestResultDao().add(testResult);
    //跳转到分数查询页面
    out.println("<script language=\"javascript\">");
    out.println("alert(\"试卷提交完成!\");");
    out.println("window.location='testResultAction.jsp'");
    out.println("</script>");
}
```

### 3.5 密码管理模块的实现

模块实现流程：

修改个人密码：进入用户个人中心->点击修改密码栏->两次输入新密码->保存，当用户修改密码时输入的新密码两次不相同，会有弹窗提示重新输入密码，其程序流程图如图 3.8 所示：

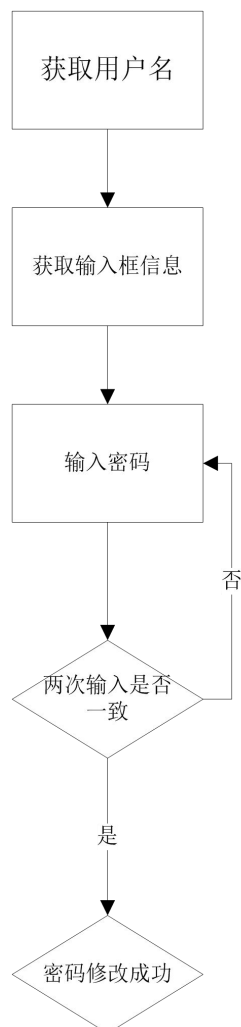


图 3.8 密码修改流程图

功能特色及技术实现：

用户登录成功后，然后点击修改密码栏，自动获取当前用户信息，可以实现对密码的修改，修改过程中，设置两次相同的新密码，即完成新密码修改，确保了安全性。实现技术利用了原密码的验证以及新密码的确认，通过输入框与后台数据比较，如果输入的原密码与后台服务器中的相同，再把输入框中新输入的密码存在后台数据中，完成了这一系列操作才算密码修改成功。

代码实现：

```
Logon logon = (Logon)session.getAttribute("admin");
```

```
//update
```

```
String password = request.getParameter("password");
```

```
if(password!=null){//判断密码是否为空
    if(logon.getPassword().equals(password)){
        //没有修改
    }else{
        MD5 m = new MD5();
        password = m.getMD5ofStr(password);
        logon.setPassword(password);//设置新密码
        AdminInfoUtil.setAdmin(logon.getUsername(), logon.getPassword());
        session.setAttribute("admin", logon);
    }
    <script language="javascript">
        alert("修改成功");//成功提示
    </script>
}
<script language="javascript">
function check(){
if(document.forms[0].password.value!=document.forms[0].confirmPwd.value){
    alert("两次密码输入不一致");
    return false;
}
    document.forms[0].submit();
}
</script>
```

### 3.6 分数查看的实现

当用户点击分数查看时，系统先获取用户的信息，把用户 id 和试卷 id 存放到 list 列表中，并把请求转向 test\_result\_view.jsp 来响应浏览器的查询请求。在 test\_result\_view 页面则通过循环列出当前用户已考试题的考试科目、考试日期和考试分数。

分数查看界面如图3.9所示。

CX在线考试系统			
<b>功能列表</b> 考试 分数查看 密码修改 退出	已考试卷		
	考试科目	考试日期	考试分数
	math	2015-04-16 12:43	50.0
	english	2015-04-16 12:45	33.3333
	math	2015-04-16 12:43	25.0
	math	2015-04-16 12:43	0.0
	english	2015-04-16 12:45	0.0

图 3.9 分数查看界面

核心代码实现如下所示：

```
//用户查询自己的考试信息

TestResultDao testResultDao = new TestResultDao();

UserInfo user = (UserInfo)session.getAttribute("user");//取得用户

int tId = 0;

if(request.getParameter("tId")!=null){

    tId = Integer.parseInt(request.getParameter("tId"));

}

int userId = user.getId();

List list = testResultDao.getReusltList(tId,userId);

request.setAttribute("list", list);//结果

request.getRequestDispatcher("/user/test_result_view.jsp").forward(request,
response);

List list = (List)request.getAttribute("list");

DateTools dateTools = new DateTools();

for(int i=0;i<list.size();i++){//通过循环列出

    TestResult testResult = (TestResult)list.get(i);//从 list 表中取出成绩

<tr>

    <td align="center"><%=testResult.getTestPaper().gettName() %></td>

    <td>
```

```
align="center"><%=dateTools.convertDateToString(testResult.getTestPaper().getStart  
Time(), "yyyy-MM-dd HH:mm") %></td>  
    <td align="center"><%=testResult.getMark() %></td>  
</tr>  
}
```

## 第 4 章 总结

虽然项目并没有实现前期设想的所有功能，但也算是实现了在线考试系统的基本功能。在总体上使用 myeclipse+mysql+tomcat 实现，保证了跨平台的能力。使用了 MVC 三层架构，使得维护更加方便。在界面上采用 div+css 布局时，使用的%进行相对定位，应用了网页自适应的思想。学习并应用了 JSP 的 jdbc 接口，也加深了对数据库的管理和控制能力，也让我对 linux 的 centos 服务器上 web 程序的发布有了实现的机会。但是时间仓促还有不少基本需求没有实现，确实还有不少地方有待改进。如：前台界面可以使用 Bootstrap（Web 前端 CSS 框架）开源框架，因为 Bootstrap 是基于 HTML5 和 CSS3 开发的，Bootstrap 中包含了丰富的 Web 组件，可以快速搭建一个漂亮、功能完备的网站。还应该添加 email 确认，手机短信验证，这样更能防止被恶意注册程序大量注册，加大服务器负担，方便添加密码找回功能来保证用户账户安全。当让也应该做 seo 优化，使得网站对搜索引擎友好，使得排名靠前。为了加大系统与用户之间的互动性和粘连度，增加学习的趣味性，应该添加评论功能和收藏功能，因为能够收藏题目，对学生用户的学习帮助很大。通过这个简单的项目，我发现了自己动手能力的不足，让我对书籍的学习有了新的理解。



## 参考文献

- [1] Peter Lubbers, Brian Albers, Frank Salim. Pro HTML5 Programming [M]. USA: Apress, 2010.
- [2] Mark Pilgrim. HTML5 Up and Running [M]. USA: O'Reilly Media, 2010.
- [3] Rob Hawkes. HTML5 Canvas: For Games and Entertainment [M]. USA: friends of ED, 2011.
- [4] Eric Plotnick. Concept Mapping: A Graphical System for Understanding the Relationship Between Concepts [EB/OL]. <http://ericit.org/digests/EDO-IR-1997-05.shtml>, 2002.
- [5] 高晓黎, 韩晓霞. SQL Server 2008 案例教程 [M]. 北京: 清华大学出版社, 2010.
- [6] CSS 权威指南 (第三版) [M]. 北京: 中国电力出版社, 2007.
- [7] 施米特. CSS cookbook 中文 (第二版) [M]. 北京: 电子工业出版社, 2007.
- [8] (美) Karl E. Wiegers 著 刘伟琴, 刘洪涛译. 软件需求 (第二版) [M]. 北京: 清华大学出版社, 2011.
- [9] 李平. 基于 JSP 的动态网页开发技术 [J]. 微计算机信息, 2009, (21): 114-116.
- [10] David Flanagan. JavaScript 权威指南 [M]. 上海: 机械工业出版社, 2007.
- [11] Jeremy Keith. JavaScript DOM 编程艺术 [M]. 北京: 人民邮电出版社, 2006.
- [12] 曹力, 张欣. JavaScript 高级程序设计 [M]. 北京: 人民邮电出版社, 2006.
- [13] 月影. JavaScript 王者归来 [M]. 北京: 清华大学出版社, 2008.
- [14] 刘彦博. 高性能网站建设指南 [M]. 北京: 电子工业出版社, 2008.
- [15] Simon Collison. CSS 基础教程 [M]. 北京: 人民邮电出版社图灵教育, 2007.
- [16] 张海藩. 软件工程导论 (第四版) [M]. 北京: 北京清华大学出版社, 2007.
- [17] 张超. 基于 JSP 的数据库连接技术浅析 [J]. 福建电脑, 2013, 28(12): 80-81.
- [18] 软件工程导论 (第四版) [M]. 北京: 北京清华大学出版社, 2007.
- [19] 杨金花. JSP 技术中文乱码的原因及解决方法 [J]. 电子设计工程, 2011, (01): 31-34.