

# 计算机信息技术课程在线考试系统的设计与实现

金 强<sup>1</sup>, 王 亮<sup>2</sup>, 方春华<sup>1</sup>

(1.江苏航空职业技术学院文理学院, 江苏 镇江 212000;

2.江苏航空职业技术学院工程学院, 江苏 镇江 212000)

**摘 要:** 本文详细地阐述了基于PHP、Mysql、Smarty技术的大学计算机信息技术课程在线考试系统的设计与实现, 作者以一次正式考试的流程介绍了系统数据库中主要几个数据表的设计和作用, 剖析了在线考试组卷的方法及过程, 阐述了从考生登录、考试页面生成、考试中页面的检查、试卷的提交、答案的保存、考试的评分到成绩的查询等全部过程的设计方法, 以图文并茂的方式介绍了系统前、后台页面功能设计, 并对整个考试系统的性能进行了测试与完善。

**关键词:** 计算机信息技术; PHP; Mysql; Smarty; 在线考试

**中图分类号:** TP311.5 **文献标识码:** A

## Design and Implementation of the Online Examination System for Computer Information Technology Courses

JIN Qiang<sup>1</sup>, WANG Liang<sup>2</sup>, FANG Chunhua<sup>1</sup>

(1.College of Arts and Sciences, Jiangsu Aeronautical Vocational and Technical College, Zhenjiang 212000, China;

2.School of Engineering, Jiangsu Aeronautical Vocational and Technical College, Zhenjiang 212000, China)

**Abstract:** This paper elaborates on the design and implementation of the online examination system for university computer information technology courses based on PHP, Mysql and Smarty. It introduces the design and function of several main data tables in the system database by means of a formal examination process, analyzes the method and process of examination paper generation, and expounds the design method of the whole process from the candidate's login, examination page generation, examination page check, examination paper submission, answer saving, examination paper scoring, and score query, etc. The function design of the front stage and back stage is illustrated, and the performance of the entire examination system is tested and improved.

**Keywords:** computer information technology; PHP; Mysql; Smarty; online examination

## 1 引言(Introduction)

当今, 计算机与信息技术的应用已渗透到几乎所有的学科和专业, “计算机信息处理”已成为高等学校普遍开设的公共基础课<sup>[1]</sup>, 江苏省每位大学生都需要参加省计算机一级统考, 但由于省考练习系统机考理论部分没有评分系统, 学生在做了理论题后不知道对错, 更没有答案解析, 因此有必要设计一个考试系统, 把历年考试的理论试题, 按照题型、分值、答案、必要的解析和最后的评分融合在一起, 用这个系统对学生进行理论知识的强化训练, 使学生较好地掌握理论知识、通过考试, 随着互联网技术的不断发展, 基于Web的在线考试系统应运而生<sup>[2]</sup>, 从开发技术的角度进行划分, 目前的web网页可以分为前端开发和后台的开发<sup>[3]</sup>, 本系统基

于PHP、Mysql和Smarty技术, 采用B/S模式, 前后台通过PHP和Smarty模板进行数据库操作和页面的生成, 能够实现在线考试及评分。

## 2 在线考试系统的设计(Design of online examination system)

### 2.1 数据库设计

数据库名为db\_online, 它主要包含以下数据表: 用户表(tb\_user)、正式考试考生表(tb\_examination\_user)、正式考试试卷表(tb\_examination)、模拟考试考生表(tb\_mockexaminations\_user)、模拟考试试卷表(tb\_mockexaminations)、在线考生表(tb\_onlineuser)、课程分类表(tb\_types\_course)、卷面题型分配模板表

(tb\_problemtemplate)、卷面总分、及格分数线表(tb\_examattribute)、试题表(tb\_exam\_problem)、试卷题型表(tb\_exam\_type)、难度系数表(tb\_degree)、单项练习表(tb\_useranswer)它们之间的关系如下图所示:

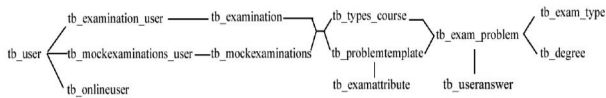


图1 数据表关系

Fig.1 Data table relationship

图1中所有表的text类型和varchar类型的字段编码均为utf8\_unicode\_ci,数据库存储引擎为MyISAM,下面以计算机基础课程的一次正式考试流程来介绍几个主要数据表(以下简称试题表、试卷表、模板表、考生表)的内容:管理员在后台使用试题表(包含'id','content','search\_list','answer','fraction','pro\_type','pro\_class','upload\_date','resolve','isactive','degreeid','sound','mustknown'字段)手工添加试题,表中'id'字段是每条试题的试题号,int类型,自动增长,具有唯一性,'content'字段填写试题内容,text类型,'search\_list'字段填写每种题型的选项个数,整型,本课程单选题(radio)、多选题(checkbox)个数取4,判断题(judgment)和填空题(fill)取0,'answer'字段填写每题正确答案,text类型,'fraction'字段填写每题分值,int类型,'pro\_type'字段填写题型,varchar类型,'pro\_class'字段填写课程类别,int类型,'upload\_date'字段填写试题添加或更新日期,date类型,'resolve'字段填写试题解析,text类型,'isactive'字段填写试题是否启用或禁用,varchar类型,'degreeid'字段填写难度系数,int类型,对应难度系数表中的较易、中等、较难3个选项,'mustknown'字段填写是否必须掌握,int类型,默认为1(必须掌握),'sound'字段填写语音上传文件(可选),varchar类型。由于信息技术课程只有单选、判断和填空三种题型,因此试题表中无需添加多选题,当数据库中添加了一定数量不同题型的试题后,管理员便可随机生成试卷表(包含'id','radio','checkbox','fill','judgment','title','pro\_class','exam\_user','start\_exam','over\_exam','dates','templateid'字段),表中的'id'字段填写的是生成的试卷号,int类型,自动增长,具有唯一性,'radio','fill','judgment'三个字段分别填写的是由随机选出的3种不同题型的试题号组成的字符串,text类型,每个试题号对应于试题表的'id'字段值,每种题型由试题号之间的@符号连接而成,譬如'radio'字段:9475@9493@9494@9495,数字9475代表试题表中某条试题的'id'值,而非试题内容,字符串的长度或者准确说每种题型的个数依据'templateid'字段值,'title'字段填写试卷标题,text类型,'pro\_class'字段填写课程类别,int类型,'exam\_user'字段填写参加考试人员,text类型,其值是一组由@符号组成的字符串,譬如90@88@89,数字90表示用

户表中的'id'用户,'start\_exam'字段填写开考准确时间,'over\_exam'填写考试结束时间,两个字段均为varchar类型,'dates'填写试卷生成的日期,date类型,'templateid'字段填写试卷模板,int类型,其值等于模板表(包含'id','name','emattrid','typedistribute','degreedistribute'字段的'id'字段值,譬如信息技术课程的'templateid'值为29,其值就是模板表id为29的值,管理员在随机生成试卷时,首先要选择课程类别和试卷模版,试卷模板依据模板表的'typedistribute'字段,譬如信息课程的模板为radio@50@checkbox@0@fill@20@judgment@30,表示单选题50分,填空题20分,判断题30分,无多选题。模板表中'id'字段表示试卷模板号,int类型,自动增长,具有唯一性,'name'字段填写模板名称,varchar类型,'emattrid'字段填写与总分、及格分数线表关联的'id'值,int类型,'typedistribute'字段填写题型分配,'degreedistribute'字段填写难度系数分配,二者均为varchar类型。每场考试完毕,系统要把考试结果填写到考生表(包含'id','title','examination\_id','name','admission','pro\_class','fraction','sub\_answer','sub\_time','nameid'字段)中,表中'id'字段填写考试编号,int类型,自动增长、唯一,'title'字段填写试卷标题,varchar类型,'examination\_id'字段填写试卷'id'号,int类型,其值与试卷表的'id'相关联,'name'字段、'admission'字段,分别填写考生姓名、准考证号,varchar类型,'pro\_class'字段填写课程类别,'fraction'字段填写成绩总分,int类型,'sub\_answer'字段填写考试答案,text类型,其值是形如\*9541@radio@A@A@1\*9491...@judgment@正确@正确@1\*...\*9517@fill@路由器@路由器@1\*...的一个长字符串,每条试题答案之间用“\*”连接,'sub\_time'字段填写试卷提交日期和时间,datetime类型,'nameid'字段填写考生'id',int类型,其值与用户表的'id'相关联。

## 2.2 组卷过程设计

组卷时,系统首先通过试卷模板得到题型和数量,然后通过随机函数抽题,如果抽题成功,则将生成的试卷号'id'值添加到数据库试卷表中,通常情况下,每个课程的考题数量要大于试卷模板要求的考题数量,譬如信息技术课程单选题有500多条(当然我们可以通过启用和禁用字段进行选择),而试卷模板单选题只考50条(每题1分,也可每题2分25条),因此随机抽题时,抽题的范围是500条,而抽到试题的数量由模板表下此题型的数量决定,假如500条单选题通过禁用,剩下30条,如果还是每条1分,显然题数不够,不能成功抽取,如果每题2分,那就在30条里面抽取25条,试题越多,分散度就越大,以单选题抽取为例,系统使用array\_rand(\$radios,number)函数随机从试题表中取得number数量个单选题键名(即ID值)组成的数组,通过for语句循环读取获取的键值,然后根据键值从试题表中获取指定'id'的试题,并

添加到试题数组中,最后通过`$radio=implode('@',$radio_exam)`,对生成的试题数组进行由数组到字符串的转换,以`@`为分隔符,形如`9475@9493@9494@...@9689`,最后将变量`$radio`通过SQL插入语句填写到试卷表对应的`'radio'`字段中,字符串`@`的个数即为单选题的数量,系统获得所有题型和数量之后,将试卷标题、考试时间一并添加进试卷表,组卷成功。

### 2.3 考试和评分过程设计

下面以正式考试情景介绍考试和评分过程的设计,考生在前台页面选择正式考试单元,系统弹出登录窗体,考生输入准考证号、密码,在下拉列表中选择考试名称,点击登录按钮,开始进入考试,这里要说明的是考试名称除了教师在后台生成的名称之外,考生还可以选择随机生成考卷,系统根据考生的准考证号,查询用户数据表,取得考生的课程类别,根据课程类别和该课程类别下一个固定的试卷模版,生成名为随机考试的试卷,其抽题原理与后台管理员组卷过程基本相同。考生登录时的准考证号和密码如果和用户表中一致,系统便会读取试卷表中的开考时间,如果登录时间与开考时间相同,则开始考试,如果开考时间未到,系统会弹出考试时间未到并显示等待时间窗口,如果考试时间已超过15分钟,会弹出已超时窗口。考生登录成功后,系统会将考生`id`、试卷`id`、准考证号、姓名、开考科目等信息存储到服务器的Session变量中,然后系统根据Session变量中的试卷`id`值,去数据库试卷表中查询得到试卷标题、开考时间和结束时间,并将上述时间分别转换为unix时间戳,存储到服务器Session变量中,同时根据两个unix时间戳的差值计算考试总时间,具备以上条件,则可生成考试页面。

生成的考试页面应包含考生姓名、准考证号、考核类型、试卷标题、题型、数量、分值、考试计时、剩余时间,以及每条具体试题内容,由于存储在试卷表中的试题是按题型分类的试题题号,而不是试题内容本身,因此要根据试卷表中的每个题型的`id`值,去试题表中将具体试题内容查询出来,下面以单选题(`radio`)为例,分析如何根据试题题号得到具体试题内容的过程:系统首先根据试卷的`id`值,通过`$radio=explode('@',$kst['radio'])`函数读取试卷表中单选题(`'radio'`字段)中的数据,将形如`9475@9493@9494@...@9689`样式的字符串拆分成数组,然后通过for循环语句读取数组中每条单选题的`id`,根据此`id`值向试题表查询得到试题内容,譬如题号`id`为`9475`,则可在试题表中将`id`为`9475`的试题内容(`'content'`字段)读取出来,用同样的方法可取得所有题型的考试试题。考生进入考试后,系统运用Ajax技术,无刷新实现在考试页面头部显示考试计时时间和剩余时间,考试剩余时间由SESSION变量中的考试结束时间值减去当前时间的unix时间戳得到,而计时开始时间值为当前unix时间戳减去SESSION变量中的开考时间,系统运用jQuery Ajax的post方法,利用回调函数来实现考试期间已答和未答试题的统计显示,采用数组`$sub_answer=array()`存储考生答案,采用变量

`$fraction`存储分数,当计时时间到或者考生提前交卷,系统获取表单提交的数据,计算并保存考试结果,以题号`9541`单选题答题为例:系统首先判断有无单选题题型存在,如果存在,则将考生提交答案与试题表中`id`为`9541`试题的正确答案进行比较,如果提交答案与正确答案相同,则按照试题表下每题分值(`'fraction'`字段)大小进行加分,如果答错或者未答则给予0分,通过for循环语句遍历所有单选题,将形如`9541@radio@A@A@1`的字符串结果存入`$sub_answer[]`数组中,同时计算单选题总分,当所有题型遍历完后,计算试卷总分,并把考试结果数据填写到考生表中,表中考生答案(`'sub_answer'`字段)用`implode("",$sub_answer)`函数拼接成字符串,如果填写成功,表示完成试卷信息的提交,系统弹出试卷提交成功信息,并同时显示成绩结果页面。

考生也可以在考试结束后根据准考证号和试卷名称查询成绩,显示成绩结果页面。系统利用`$ar=explode("",$sub_answer)`函数,拆分得到每条试题的考试结果,利用for循环读取每条试题的数据,利用`$dates=explode("@",$ar[$i])`将每条试题的数据再拆分到数组中,利用`$array_type[]=$dates[1]`将题型存储到数组中,利用`$answers[]=$dates[2]`将正确的答案存储到数组中,利用`$answer[]=$dates[3]`将提交的答案存储到数组中,利用`$array_answer[]=$dates`将每条试题的所有数据存储到数组中。由于成绩页面中要显示具体的试题内容、正确答案、提交答案,因此,系统根据试卷`id`值,将试卷的题型先放到变量`$examtype`中,通过foreach语句遍历该题型下的所有试题`id`值,然后根据试题`id`值将试题内容从试题表中取出来,付给页面模板。在页面模板中,以单选题为例,系统利用双重循环语句进行遍历,遍历的条件就是考生表中`'sub_answer'`字段包含的试题`id`、题型分别与试题表中的试题`id`、题型相等,遍历结果中的试题内容取自试题表中的`'content'`字段、正确答案取自试题表中的`'answer'`字段、提交答案取自考生表中的`'sub_answer'`字段、问题解析取自试题表中的`'resolve'`字段。页面中,考生还可以通过单击查看答题情况按钮查看所有答题情况,单击正确或错误按钮,页面会跳转到对应试题所在位置实现单条试题的结果查询。

### 2.4 系统前后台功能设计

客户端程序在功能设计上共分为前台和后台两大部分,前台设计主要包括首页设计,以及单项练习、模拟考试、正式考试、成绩查询、新闻公告五大模块入口设计,如图2所示。由于系统采用了Smarty模版引擎技术,它将应用程序分成视图和逻辑控制两部分<sup>[4]</sup>,有效地实现了逻辑代码与内容表现的分离,专注于内容表现,避免了标签与程序的混合,使得逻辑内容的改变不会影响到前台视图<sup>[5]</sup>,因此大大减轻了页面设计的工作量,每个模板页面通过对应PHP文件的`$smarty->display`方法指定,页面的变量通过对应PHP文件的`$smarty->assign`方法指定。在单项练习模块的页面设计中,运用了Jquery的Ajax技术,对提

交按钮进行判断, 构建function checkInput(id,type)函数对单选题、判断题、多选题进行题型检查, 构建function checkAnswer(id,type)函数对是否选择了答案进行检查, 构建function createInputHidden(id)函数创建用户已答试题的隐藏域, 记录用户所答的问题。与模拟考试和正式考试不同的是, 单项练习模块的答案是提交到当前页面, 在隐藏域里利用setAttribute方法创建input元素的name值'hid', 根据\$\_POST['hid']方法到数据库试题表中查询该id试题的正确答案, 将提交的答案与之比较, 正确则加分, 然后将提交答案和正确答案输出到当前页中, 用户每答一题, 就向单项练习表中添加一条记录, 保存用户所做的数据。模拟考试模块与正式考试模块的功能设计基本相同, 区别只在于模拟考试使用的是测试账号, 登录时用户只需选择考题名称而无须输入准考证号和登录密码。

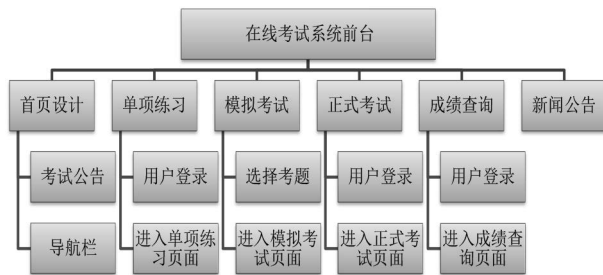


图2 前台功能

Fig.2 The foreground function

后台页面设计采用了框架布局的形式, 管理员在后台进行登录时, 系统首先运用JavaScript对登录账号、密码、权限是否填写进行检查, 然后去数据库用户表验证所填写的账号、密码是否正确, 账户是否被冻结, 验证通过后进入后台管理系统首页, 后台首页分为头部、左部导航栏、右部功能展示区, 头部主要显示登录账户、用户权限和上次登录时间, 导航栏包含所有功能菜单, 它使用JavaScript实现子菜单的展开和折叠, 所有功能如图3所示, 其中类别管理菜单既可以添加主课程类别, 也可以添加该课程下的子课程类别, 添加考生信息子菜单中的添加考生方法既可以使用富文本编辑器CKeditor手工逐条添加, 也可以用Excel文件批量导入, 新闻公告菜单主要实现考试信息及相关新闻的发布, 试题管理菜单主要用于试题的增删改查, 试题除了可以手工逐条输入之外, 也可以进行批量导入, 导入前需分别为四种题型建立对应的Excel文件, 导入时在页面先选择课程类别、题型、题型分值, 然后选择要导入的文件, 文件格式要求单选题和多选题应包含序号、试题内容、A—D选项内容, 答案(大写字母A—D)、解析等字段, 判断题和填空题应包含序号、试题内容、答案(判断题用数字1表示正确、0表示错误)、解析等字段, 除了导入功能, 试题还可以导出为word文档, 方便打印纸质试卷。试卷管理菜单下的分数和分数线设定子菜单, 可以设置总分和及格分数线, 按照总分和分数线可以在模板定义子菜单中设定不同题型的试题数量, 并保存为新模板。在

模板管理子菜单中可以修改各个题型的数量占比, 在管理题型子菜单中不但可以更改或删除现有题型, 而且还可以增加其他非客观题型, 进行人工阅卷。成绩管理菜单可以用于正式考试成绩的查询和统计, 在使用成绩查询子菜单进行查询时, 查询的选项可以是考生姓名、考试名称、课程类别和考试不及格者, 可以将查询的结果导出为Excel格式的文件。在成绩统计子菜单, 系统利用JpGraph图表类库实现某个考试名称的成绩统计与分析, 利用直线图表示所有考生的考试成绩, 利用3d饼图表示考试通过率。权限管理菜单可以实现从用户表中添加用户为教师、管理员或超级管理员角色, 也可以对已添加的角色进行权限更改, 根据管理权限的不同, 用户以教师身份角色登录后台时导航栏没有权限管理菜单, 以教师和管理员角色登录后台时导航栏含有数据库管理菜单, 可用于对考试数据库进行备份或还原, 保证考试数据的安全。



图3 后台功能

Fig.3 The background function

### 3 系统测试和完善(System testing and improvement)

实现系统的稳定性是运行在线考试的关键<sup>[6]</sup>, 系统使用PHP技术使得考试数据得以及时更新, 但频繁的数据读取却给服务器带来了巨大的压力, 特别是当并发访问<sup>[7]</sup>量较大时, 服务器频繁工作于用户请求与数据库响应两个操作之间, 就会造成用户请求多与数据库响应慢两者之间相互等待的局面, 严重者则会导致系统崩溃。相对于其他模板引擎来说, 采用Smarty模板编写的应用程序可以获得更大效能, 在应用程序第一次运行时, Smarty会将程序编译成一个非模板格式的PHP文件, 并将最终输出的内容缓存成一个静态的HTML文件, 若在源代码未变动或缓存时间尚未过期的情况下, 下次再访问模板时, Smarty会将用户请求直接定位到这个HTML文件上, 而不再进行重编译, 既省去了业务逻辑<sup>[8]</sup>处理麻烦, 又降低了访问量较大引发的服务器工作负荷, 有效提升了应用程序的性能和服务器的响应速度。系统在完成了整个考试系统的前后台设计之后, 必须对系统的稳定性和并发能力进行测试, 测试的首要项目是服务器单机测试, 系统在加载Apache服务和Mysql服务之后, 检查前台登录各个单元模块是否流畅, 是否存在服务器端口冲突, 后台试题导

(下转第58页)