

选题系统

1 开发背景及意义

随着计算机技术的飞速发展，计算机在各行各业中的应用迅速普及，计算机的应用在线答题中同样也得到了飞速的发展。对于选题系统最基层的管理之一在线答题来说，利用计算机实现在线答题等日常工作来提高办事效率，是适应当今社会发展要求，推动在线答题领域现代化快速发展的必要条件。虽然在线答题只是选题系统的一个环节，但是在线答题是一项琐碎、复杂而又十分细致的工作。试卷答题浪费了大量的纸张，不利于重复使用，而利用计算机就不同了，计算机可以在线出题，随机生成考题，不会浪费纸笔。同时计算机具有纸答题所无法比拟的优点。例如：答题结束即可知道排名成绩、可靠性高、存储量大、保密性好、寿命长等。这些优点能够极大地提高答题的效率，达到事半功倍的效果。不同地方的学校具有不同的试题，这也决定不同的地方有不同的出题，对于我校理学院来说目前就是这个情况，面对实际，迫切需要开发出一个新系统来适应这些工作。

本系统是结合了理学院的实际情况，进行综合考察，经过实际的需求分析采用了功能强大的 PHP 作为开发工具，用 Mysql 作为数据库，开发出的选题系统。整个系统从符合操作简便、界面友好、灵活、实用、安全，完成学生注册、答题，老师出题、删题、改题，经过实践证明本系统完全可以满足这个方面的需要。

师资力量的评估是衡量一个学校教学质量高低与否的重要标准，但它所涉及的内容比较多而且复杂，因此做出一套及时完整有效的选题系统，即方便出题、方便答题、方便查看排名。

选题系统利用了方便的网络通信技术，使用当前流行的 PHP 网络编程技术，通过网上出题，网上答题，使得对学生知识掌握情况有一个更直观的了解。

2 系统分析

在系统开发之前要进行前端分析来确定系统中的内容显示方式以及系统功能结构。下面前期分析主要包括需求分析、功能分析和技术要求三方面。

2.1 需求分析

本系统主要是针对各高校，旨在实现更好在线测验系统化管理。经过对选题系统的系统分析，有以下几方面的要求：

- (1)教师布置书面作业不容易验收, 批改浪费时间, 在线答题通过计算机在线判正误速度快。
- (2)排名系统直观的显示学生的排名情况。
- (3)投入成本低;由于该系统的生命周期不是很长, 功能有限, 工作范围小, 数据量不是很大, 因而投资也不是很大。

2.2 功能分析

功能可以说是一个系统的后勤支持, 系统形象、内容规划的表现好坏很大程度上在于功能开发上面, 功能开发的质量优劣与多寡也反映出一个学校实力的强弱, 好的系统不是简单的套用功能模块, 而是根据学校的特点, 融入自己的思想, 有针对性地开发出功能强大更具人性的功能系统。根据选题系统的基本要求和设计原则, 通过对市场需求和用户特点的分析, 系统应具备如下功能:

1.用户登录

用户登录是为了保证系统的安全性, 只有凭借用户名、密码, 才能进入系统。不同身份的用户对应着不同的功能。

2.学生答题系统

题型分为单选题、多选题、判断题三种题型, 学生可以查看所有试题, 同时也可以进行考试, 100道题, 得分录入排名系统。

3.排名系统

学生/老师都可以进行排名查看, 排名根据学生答对题目数的多少进行排名, 答对题目数多者排名越高。

4.教师出题系统

老师选择题型(单选、多选、判断)进行出题, 并提交到数据库, 教师可以管理自己出的题, 进行修改, 删除。

5.注册系统

开放学生用户注册, 教师注册没有开放, 只能后台添加, 以保证数据安全。

3 系统总体设计

基于以上的前期分析, 初步了解了系统功能和技术要求。接下来就需要对系统进行总体设计, 大体确定为系统功能模块、数据库设计和界面设计。

3.1 设计任务

在需求分析阶段, 会形成相应的需求文档和系统的初步模型, 这些是系统设计的依据。系统设计就是根据需求文档用比较抽象概括的方式确定系统如何完成预定的任

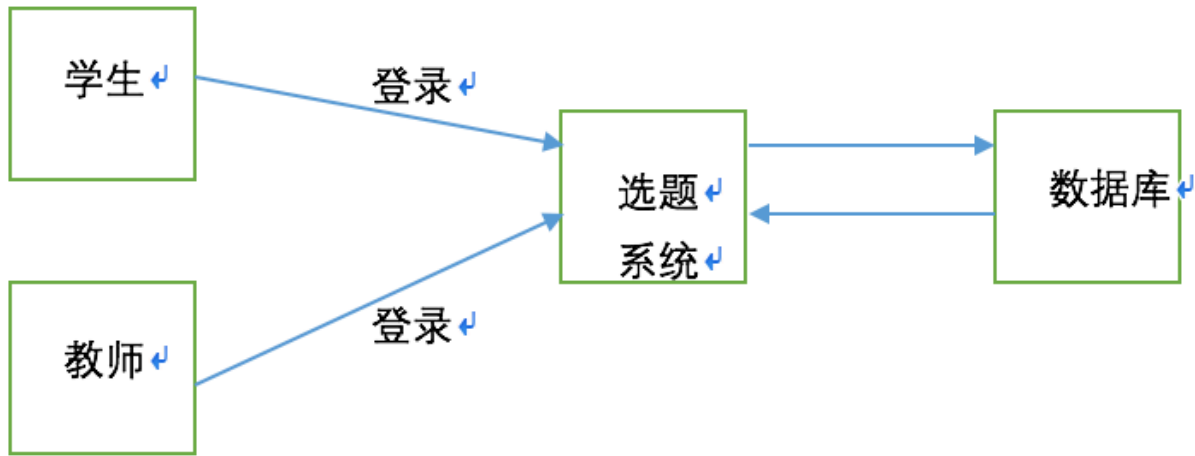
务，也就是说，应该确定系统的物理配置方法，并且进而确定组成系统地每个程序的结构、数据库的详细设计、编码设计、输入/输出界面(人、机界面)设计等。

3.2 系统功能设计

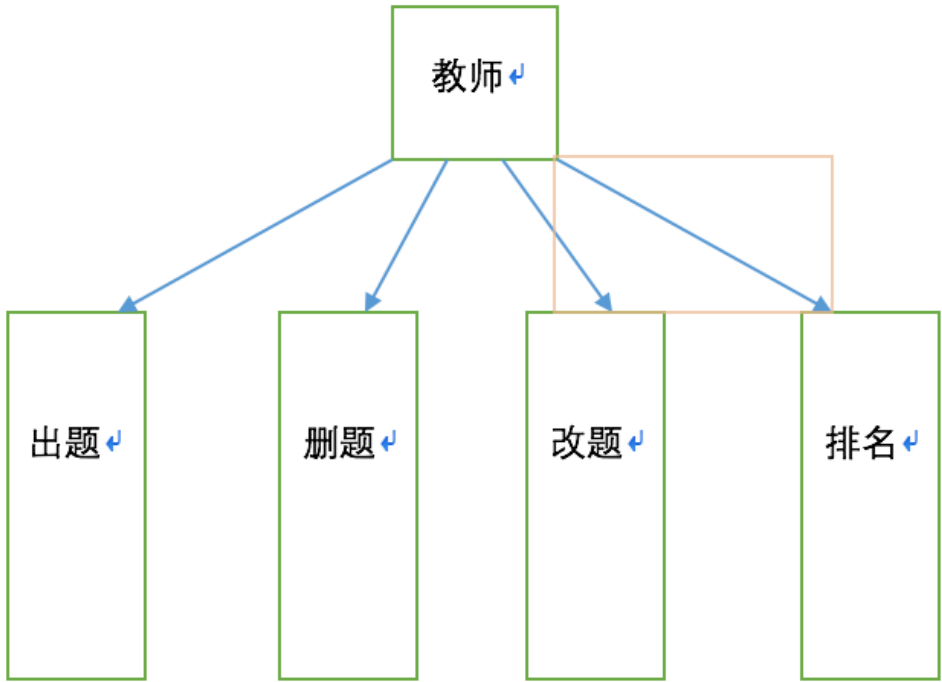
选题系统主要分为两大部分：学生登录部分，教师登录部分。

1.系统整体设计

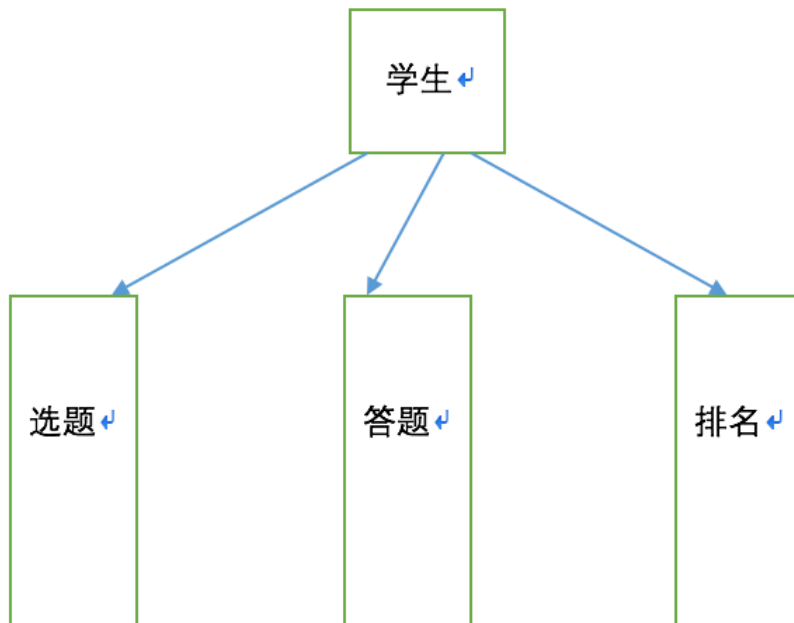
系统整体分为教师和学生两大登录模块，以不同身份登录后进行不同的操作，最终作用于数据库和用户终端。用例图如下：



教师使用系统



学生使用系统



3.3 数据库设计

数据库是计算机领域中最重要技术之一。目前，它正在迅速的发展着，特别是在应用的实践中不断丰富着新的内容。数据库是信息管理系统中至关重要的一个组成部分，它为管理信息系统存贮和管理有关的数据。因而，它是信息系统开发中一项重要工作。由于数据库技术的使用，给研制新型的各类信息系统以有力的支持。

本系统采用Mysql 5.6建立数据库，数据库名webwork，共采用2个基本数据表。

1.用户信息表。

#	名字	类型	排序规则	属性	空	默认	额外
1	id	int(10)		UNSIGNED	否	无	AUTO_INCREMENT
2	user_name	varchar(256)	gbk_chinese_ci		否	无	
3	user_password	varchar(256)	gbk_chinese_ci		否	无	
4	time	datetime			否	CURRENT_TIMESTAMP	
5	types	varchar(10)	gbk_chinese_ci		否	N	
6	standing	int(10)			否	无	

说明 user_name 用户名

user_password 用户密码

time 注册时间戳

types 用户类型

standing 学生排名

2.题库表

#	名字	类型	排序规则	属性	空	默认	额外
1	id	int(10)		UNSIGNED	否	无	AUTO_INCREMENT
2	problem	varchar(4096)	gbk_chinese_ci		否	无	
3	pro	varchar(4096)	gbk_chinese_ci		否	无	
4	daan	varchar(4096)	gbk_chinese_ci		否	无	
5	type	varchar(10)	gbk_chinese_ci		否	P	
6	who	varchar(10)	gbk_chinese_ci		否	无	

说明 problem 完整问题描述

pro 去答案问题描述

daan 问题答案

type 题型

who 出题者

3.4 界面设计

一个好的系统应该有非常好看的设计、友好的界面、毫无缺点的结构、智能化的后台处理、非常优秀的内容以及优良快捷的交互性。即系统的建设具有交互性与持续性、艺术与技术结合的紧密性等特点。系统的界面设计在 e 时代十分重要。系统界面设计的三大原则是:置界面于用户的控制之下;减少用户的记忆负担;保持界面的一致性。界面设计在工作流程上分为结构设计、交互设计、视觉设计三个部分。结构设计 Structure Design 是界面设计的骨架;交互设计 Interactive Design 的目的是使系统用户能简单使用,如操作错误时,系统提供针对性的提示。

系统界面应该能为用户服务,是用户与计算机沟通的唯一途径。一般来说系统界面设计应尽量做到:

- 1.简易性。界面的简洁是要让用户便于使用、便于了解、并能减少用户发生错误选择的可能性。
- 2.一致性。这是每一个优秀界面都具备的特点。界面的结构必须清晰且一致,风格必须与内容相一致。
- 3.清楚。在视觉效果上便于理解和使用。
- 4.排列。一个有序的界面能让用户轻松的使用。
- 5.安全性。用户能自由的作出选择,且所有选择都是可逆的。在用户作出危险的选择时有信息介入系统的提示。
- 6.灵活性。简单来说就是要让用户方便的使用。

作为选题系统,风格上应当既美观大方,又简洁明了。本站在界面设计时充分考虑了以上几点。

1.登录界面



2.注册界面



3.教师出题系统界面



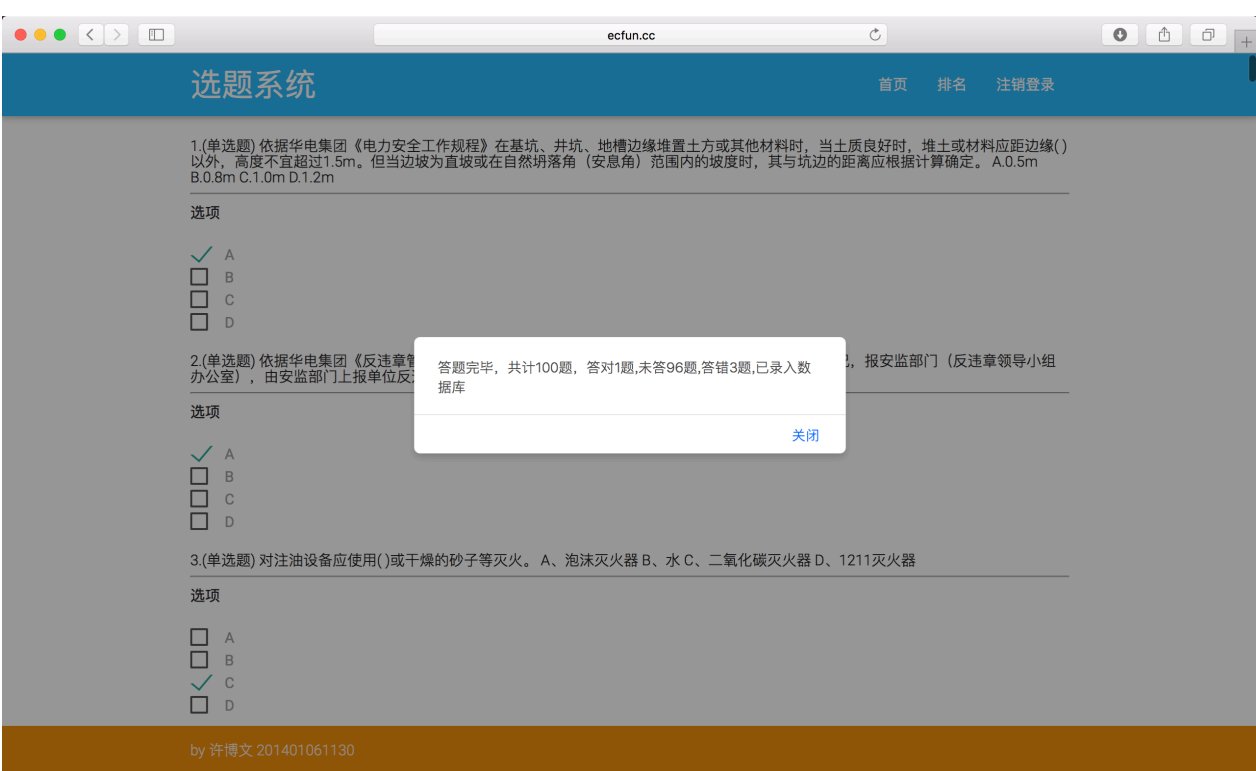
4.教师修改题目界面



5.学生系统界面



6.学生答题系统界面





7.排名系统界面



4 系统详细设计

基于以上的总体设计，进一步界定了系统功能和技术要求。接下来就 需要对系统进行详细设计，根据已经确定的总体设计以及用户需求分析， 详细设计将从系统学生登录模块、教师登陆模块两部分论述。

4.1 系统登录模块功能的实现

用户登录时输入用户用户名、密码，点击登录按钮跳入不同的页面。如果是教师用户，则跳转到教师用户页面;如果是学生用户，则跳转到学生页面。如果输入信息有误，则提示输出错误，请确认后重新填写。

后台代码

```
<?php
/**
 * Created by PhpStorm.
 * User: xubowen
 * Date: 2017/6/4
 * Time: 下午3:02
 */
require_once '../lib/config.php';
session_start();
$email = $_GET['email'];
$password = $_GET['password'];
$sql="select user_password,types,id from user_information where user_name='$email'";
$result = mysqli_query($conn,$sql);
$row=mysqli_fetch_row($result);
if($password!=null&&$email!=null){
    if($row[0]==$password){
        $rs['code'] = '1';
        $rs['ok'] = '1';
        $rs['msg'] = "欢迎回来";
        $_SESSION['username']=$email;
        $_SESSION['types']=$row[1];
        $_SESSION['id']=$row[2];
        if($row[1]=="Y"){
            $_SESSION['title']="出题系统";
        }else{
            $_SESSION['title']="选题系统";
        }
    }else{
        $rs['code'] = '0';
        $rs['msg'] = "账号或者密码错误";
    }
}else{
    $rs['code'] = '0';
    $rs['msg'] = "请填写账号或密码";
}

echo json_encode($rs);
```

前台代码


```

    </div><!-- /.login-box-body -->
</div><!-- /.login-box -->

<!-- jQuery 2.1.3 -->
<script src="asset/js/jquery.min.js"></script>
<!-- Bootstrap 3.3.2 JS -->
<script src="asset/js/bootstrap.min.js" type="text/javascript"></script>
>
<!-- iCheck -->
<script src="asset/js/ichack.min.js" type="text/javascript"></script>

<script>
    $(document).ready(function(){
        function login(){
            $.ajax({
                type:"GET",
                url:"api/_login.php",
                dataType:"json",
                data:{
                    email: $("#email").val(),
                    passwd: $("#passwd").val(),
                },
                success:function(data){
                    if(data.ok){
                        $("#msg-error").hide(100);
                        $("#msg-success").show(100);
                        $("#msg-success-p").html(data.msg);
                        window.setTimeout("location.href='index.php'",
1000);

                    }else{
                        $("#msg-error").hide(10);
                        $("#msg-error").show(100);
                        $("#msg-error-p").html(data.msg);
                    }
                },
                error:function(jqXHR){
                    $("#msg-error").hide(10);
                    $("#msg-error").show(100);
                    $("#msg-error-p").html("发生错误: "+jqXHR.status);
                    // 在控制台输出错误信息
                    console.log(removeHTMLTag(jqXHR.responseText));
                }
            });
        }
        $("#html").keydown(function(event){
            if(event.keyCode==13){
                login();
            }
        });
        $("#login").click(function(){
            login();
        });
    });

```

```

        $("#ok-close").click(function(){
            $("#msg-success").hide(100);
        });
        $("#error-close").click(function(){
            $("#msg-error").hide(100);
        });
    })
</script>
<script type="text/javascript">
    function removeHTMLTag(str) {
        str = str.replace(/<\/?[^\>]*>/g, ''); //去除HTML tag
        str = str.replace(/[ | ]*\n/g, '\n'); //去除行尾空白
        str = str.replace(/\n[\s| | ]*\r/g, '\n'); //去除多余
        //空行

        str = str.replace(/&nbsp;/ig, ''); //去掉&nbsp;
        return str;
    }
</script>
</body>
</html>

```

js ajax请求后台进行数据交互，types N为学生，Y为教师，根据types来判断用户界面

4.2 教师模块功能的实现

1.出题后台代码

```

<?php
/**
 * Created by PhpStorm.
 * User: xubowen
 * Date: 2017/6/5
 * Time: 下午10:08
 */
require_once '../lib/config.php';
session_start();
$pro = $_GET['pro'];
$daan = $_GET['daan'];
$type = $_GET['type'];
$pattern = "/\([^\\]*\) /";
$str=preg_replace($pattern, "<a>$daan</a>", $pro);
$who=$_SESSION['id'];

if($pro!=null&&$daan!=null){
    $sql="insert into ".$type."(problem,pro,daan,who) values('$str','$p
ro','$daan','$who)";
    $result = mysqli_query($conn,$sql);
    if($result){
        $rs['code'] = '1';
    }
}

```



```
$rs['ok'] = '1';
$rs['msg'] = "录入成功";
}else{
    $rs['code'] = '0';
    $rs['msg'] = "录入异常";
}
}else{
    $rs['code'] = '0';
    $rs['msg'] = "问题答案不能为空";
}
echo json_encode($rs);
```

2.修改题目后台

```
<?php
/**
 * Created by PhpStorm.
 * User: xubowen
 * Date: 2017/6/6
 * Time: 下午3:52
 */
require_once '../lib/config.php';
$pro = $_GET['pro'];
$daan = $_GET['daan'];
$type = $_GET['type'];
$id = $_GET['id'];
$pattern = "/\([^\\)]*\)/";
$str=preg_replace($pattern, "<a>$daan</a>", $pro);
if($pro!=null&&$daan!=null){
    $sql="update ".$type." set pro='$pro',daan='$daan',problem='$str' w
here id='$id'";
    $result = mysqli_query($conn,$sql);
    if($result){
        $rs['code'] = '1';
        $rs['ok'] = '1';
        $rs['msg'] = "录入成功";
    }else{
        $rs['code'] = '0';
        $rs['msg'] = "录入异常";
    }
}else{
    $rs['code'] = '0';
    $rs['msg'] = "问题答案不能为空";
}
echo json_encode($rs);
```

3.删除题目后台

```

<?php
/**
 * Created by PhpStorm.
 * User: xubowen
 * Date: 2017/6/6
 * Time: 下午3:10
 */
require_once '../lib/config.php';
$type = $_GET['type'];
$id = $_GET['id'];
$sql="delete from ".$type." where id='".$id'";
$result = mysqli_query($conn,$sql);
if($result){
    echo ' <script>alert("删除成功!")</script> ';
    echo " <script>window.location='../myquestionshow.php?type=".$type.
    "';</script> " ;
}

```

4.3 学生模块功能的实现

1.选题后台

```

<?php
/**
 * Created by PhpStorm.
 * User: xubowen
 * Date: 2017/6/4
 * Time: 下午4:52
 */
$response = array();
require("lib/db_config.php");
require("lib/config.php");

function selectRandomProblem(){//随机100题
    global $conn;
    global $danxuan;
    global $duoxuan;
    global $panduan;
    $sql1="select pro, daan, type from danxuan order by rand() limit $d
anxuan";
    $sql2="select pro, daan, type from duoxuan order by rand() limit $d
uoxuan";
    $sql3="select pro, daan, type from panduan order by rand() limit $p
anduan";
    $result1=mysqli_query($conn,$sql1);
    $result2=mysqli_query($conn,$sql2);
    $result3=mysqli_query($conn,$sql3);
    $array=array();

```

```

        if($result1&&$result2&&$result3){
            while($row=mysqli_fetch_array($result1)){
                $array[] = array(
                    'pro'=>$row[0],
                    'daan'=>$row[1],
                    'type'=>$row[2],
                );
            }
            while($row=mysqli_fetch_array($result2)){
                $array[] = array(
                    'pro'=>$row[0],
                    'daan'=>$row[1],
                    'type'=>$row[2],
                );
            }
            while($row=mysqli_fetch_array($result3)){
                $array[] = array(
                    'pro'=>$row[0],
                    'daan'=>$row[1],
                    'type'=>$row[2],
                );
            }
            return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
        }else {
            $array[] = array(
                'pro'=>'',
                'daan'=>'',
                'type'=>'',
            );
            return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
        }
    }
}

function selectDanxuanProblem(){//全部单选题
    global $conn;
    $sql1="select pro, daan, type from danxuan";
    $result1=mysqli_query($conn,$sql1);
    $array=array();
    if($result1){
        while($row=mysqli_fetch_array($result1)){
            $array[] = array(
                'pro'=>$row[0],
                'daan'=>$row[1],
                'type'=>$row[2],
            );
        }
        return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
    }else {
        $array[] = array(
            'pro'=>'',
            'daan'=>'',
            'type'=>'',
        );
    }
}

```

```

        return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
    }
}
function selectDuoxuanProblem(){//全部多选题
    global $conn;
    $sql1="select pro, daan, type from duoxuan";
    $result1=mysqli_query($conn,$sql1);
    $array=array();
    if($result1){
        while($row=mysqli_fetch_array($result1)){
            $array[] = array(
                'pro'=>$row[0],
                'daan'=>$row[1],
                'type'=>$row[2],
            );
        }
        return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
    }else {
        $array[] = array(
            'pro'=>'',
            'daan'=>'',
            'type'=>'',
        );
        return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
    }
}
function selectPanduanProblem(){//全部判断题
    global $conn;
    $sql1="select pro, daan, type from panduan";
    $result1=mysqli_query($conn,$sql1);
    $array=array();
    if($result1){
        while($row=mysqli_fetch_array($result1)){
            $array[] = array(
                'pro'=>$row[0],
                'daan'=>$row[1],
                'type'=>$row[2],
            );
        }
        return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
    }else {
        $array[] = array(
            'pro'=>'',
            'daan'=>'',
            'type'=>'',
        );
        return $array;//json_encode($array,JSON_UNESCAPED_UNICODE);
    }
}
}

```


2.提交成绩后台

```
<?php
/**
 * Created by PhpStorm.
 * User: xubowen
 * Date: 2017/6/6
 * Time: 下午2:36
 */
require_once '../lib/config.php';
session_start();
$total = $_GET['total'];
$id=$_SESSION['id'];
$sql="update user_information set standing = standing+'$total' where id
='$id'";
$result = mysqli_query($conn,$sql);
if($result){
    $rs['code'] = '1';
    $rs['ok'] = '1';
    $rs['msg'] = "提交成功";
}else{
    $rs['code'] = '0';
    $rs['msg'] = "提交失败";
}
echo json_encode($rs);
```

3.排名系统后台

```
<?php
/**
 * Created by PhpStorm.
 * User: xubowen
 * Date: 2017/6/6
 * Time: 下午12:34
 */
require("lib/db_config.php");
require("lib/config.php");
function getStanding(){
    global $conn;
    $sql = "SELECT id,user_name,standing FROM user_information WHERE ty
pes='N' ORDER BY standing DESC";
    $array=array();
    if($result=mysqli_query($conn,$sql)) {
        while ($row = mysqli_fetch_array($result)) {
            $array[] = array(
                0=>$row[0],
                1=>$row[1],
                2=>$row[2],
            );
        }
    }
}
```

```
}  
    return $array;  
}  
}
```

5 测试与结论

5.1 测试

5.1.1 测试环境

系统软件:CentOS 7;

数据库管理软件:Mysql 5.6;

系统开发平台:PhpStorm。

5.1.2 测试过程

本系统的测试过程分了两个步骤:

- 1.模块测试:测试了每个模块的程序;
- 2.确认测试:测试了整个系统的功能和性能。

5.1.3 测试结果

在确认测试阶段, 没有发现了问题。

5.2 结论

在制作过程中使用了 CSS+H5+JavaScript+PHP+Mysql实现, 用到了很多东西。

PHP这门语言使用比较方便, 好上手, 比较不足的地方就是没有实现低耦合、高内聚的系统,

扩展起来比较麻烦, 对require导入PHP资源文件使用的不好, 对一些文件的引入存在冲突,

或者路径错误情况, 没有考虑好。需要改进的还有很多。

参考文献

【1】 PHP和MySQL Web开发(原书第4版) : PHP and MySQL Web Development, Fourth Edition

【2】 HTML & CSS设计与构建网站