

红岩网校工作站老后台 开发文档

目录

1.前言

2.编写目的、背景及定义

3.总体设计与运行环境

4.基本工作流程

5.如何使用

6.文件参考

7.数据库参考

8.附：我的代码规范

9.附：修订日志

1.前言

想了好久前言该如何写，刚刚下笔的时候，写了几个字，感觉就像写编写目的一样，后来想想还是删了重新写吧，就写的随意一点。

来网校一年多了，在网校学到了很多，技术上，为人处事上都有了不少的成长吧。并不能说红岩网校改变了我多少，对我来说，最大的意义在于，她让我知道自己现在该干嘛，以后要有什么样的未来。自己的自制力不能说是特别的好，时常会变得非常懒惰，懒得去上课，懒得去吃饭，就坐在寝室里一直打游戏，一局又一局，很快一天就过去了。娱乐之后又显得格外的落寞与空虚，然后才发现，我还有任务，我还有工作要去完成。于是乎，又背上电脑，来网校学习工作。学习了一天，敲了一天的代码，带给自己的是充实感与成就感，此时，才真正清楚的意识到，这就是我的路，这就是我的未来。

现在这个时候，新来的学弟学妹们才刚刚入网校的门，还有学习很多很多，还要经历很多很多。他们将经历几次考核，最终成为网校未来的中流砥柱。

来说说老后台吧，我不是网校的开创元老，所以，不知道老后台是在一个什么样的环境下创造出来的，想起我当学员那会，考核的时候用得是老后台，时间花的最多的地方就是改老后台，把老后台改成符合当前要求的后台，数据库要改，功能要改，老后台的代码很乱，HTML 和 PHP 混排，还有那奇葩的缩进与代码格式。

后来，我的部长李新星让我去改下老后台，对它的代码进行一下整理，于是就有了这篇开发文档，在改后台的时候，也遇到了一些困惑与迷茫，特此感谢部长李新星与好友任晓康、陈俊安等人的帮助。

本文档第 2 节，第 3 节对老后台做了一些简要的介绍，第 4 节简要的介绍一下它的运作流程，老后台的运作流程真是一件复杂的事情。尽管它复杂，还是推荐去了解一下，以便更好的添加自己想要的功能模块。第 5 节便是告诉使用者如何去添加一个功能模块。第 6、7 节是属于参考部分（Reference），如果在使用老后台的过程中有一些不解与疑惑，希望这部分能够给予微薄的帮助。由于当时我改老后台的时候，最头疼的便是老后台的代码规范，我尽自己最大的努力去调整它的代码，让它看上去更好扩展，更好理解，所以有了第 8 节，这一节我简述了我的代码习惯，希望能帮助一些写代码很凌乱的同学，如果你觉得自己有良好的代码规范，当然可以完全忽略它。

最后，改过的老后台还是有很多不足之处，如果在使用中出现报错，或者一些其他的问题，请将出错的页面截图，与相应的文字说明（主要是当时的情况，做了什么操作导致这样的报错），或者您有更好的修改意见与想法，请发送至我的邮箱:cqupt.kross@gmail.com

黄劼

2012 年 11 月 12 日晚

2.编写目的、背景及定义

2.1.编写目的

编写目的在于让以后的网校人在使用老后台的时候（如果还在使用的话），能更快的入手，知道该如何添加新的功能，出现错误该如何解决。主要还是前者吧。

2.2.背景

老后台应该是由零七年零八年的学长们编写的，一年一代人，代码风格与处理方式上有很大的不同，由于这些差异，导致现在使用老后台做网站的时候，觉得非常的凌乱，可能会出现一些莫名其妙的错误，往往采用屏蔽错误的方式去解决，这并不是一个好的发展方式。

2.3.定义

老后台主要用于红岩网校工作站开发网络产品与内部培训交流使用，没有特殊情况，不对外开放源代码。

3.总体设计与运行环境

3.1.总体设计

整体上使用 html 代码与 php 代码混排的形式，入口文件 index.php（如果只有文件名，即该文件存在于根目录下），随后跳转至 admin.php，随后通过判断 SESSION 数组中是否存在以一串经过 MD5 加密的字符串作为键的值来判断是否用户登录，如果没有登录，将载入 admin/login.php 文件开始进行登录，如果已经登录了将执行 admin.php 文件剩余的内容，管理界面主要是由两个 frame 标签构成的，根据该管理员账号的权限，加载左边 frame 的操作列表(admin/left.php)，并通过 GET 数组加载主体 frame 的内容（实际上是通过 GET 数组的值去读取某个文件显示在这个 frame 中）。

3.2.运行环境

本后台运行于 Apache，Mysql，PHP 的环境中。编码为 GBK（数据库与文件编码均为 GBK）

测试环境为：

Apache2.2.22

Mysql5.0.27

PHP5.2.5

4.基本工作流程

续 3.1 所述，登录完成后，将会在 SESSION 中存一个值，这个值将会在每个页面的开头用来判断是否用户登录，这个值的在 SESSION 数组中的键是一串经过 MD5 加密的字符串，这个字符串在 inc/Administration.php 文件中，变量名为 \$keyString，由它经过 MD5 加密后作为 SESSION 数组的下标，里面存的是当前登录用户在数据库中 admin 表里的 id(int)字段的值。

在大多数页面的开头几行，都有一个 `require_once` 包含了一个 `admin/check.php` 文件，这个包含是非常必须的，事关后台的安全性。如果没有通过这个文件对登录与否的判断，将可能导致，直接在浏览器的地址栏输入某个操作的文件地址，会显示出相应的表单与操作按钮。这是非常不安全的。

左边的操作列表是通过管理员在数据库中 `admin` 表中的 `adminRights(int)` 字段来判断的，菜单里的每个操作都在数据库中的 `menu` 表中的 `rightsLevel(int)` 字段来表示需要的权限，管理员的权限数值大于菜单权限的数值时，这个操作便会显示在左边的操作列表上。

同时，也将把这个操作的链接输出到 `<a>` 标签的 `href` 中，每个 `<a>` 标签的 `target` 属性都是指向 `name="main"` 的 `<frame>` 标签，主体的 `frame` 框将会读取那个路径里的文件，把里面的 `html` 代码与 `php` 程序显示出来，以便后续的操作。

5. 如何使用

5.1. 如何配置 `mysql`

在 `inc/mysql.php` 中有后台自己使用的 `mysql` 类，如果你并不习惯它的使用，可以使用自己编写的 `mysql` 类，但是请包含根目录下的 `mysql_config.php` 文件，该文件是 `mysql` 的配置文件，只需设置好里面的变量，将里面的常量传入自己的 `mysql` 文件中使用即可。

5.2. 如何更改网站名称

当你使用这个后台的时候，一定需要将后台系统的名字改成诸如：某某网站后台系统这样，请与 `inc/conf.php` 中修改相关变量的值即可，后台菜单中有一个基本设置的菜单，并不推荐使用那个功能来更改网站的名称。（因为那个功能是通过写文件的形式，直接更改 `inc/conf.php` 文件的）

5.3. 如何添加新功能

老后台经常面对不同需求的网站，想做出一个万能后台适应不同的网站，我（们）暂时还没有这个能力，所以，常用的解决方法就是自己编写需要的功能添加到老后台中，满足需求。也因为如此，导致之前出现过各种版本的老后台（因为，每个人对老后台的改写都不一样）。

如果只是改改文章的分类，可以直接去数据库中的 `type` 表改（考虑做一个修改文章分类的内部功能）。

添加新功能：首先你需要在菜单中注册一个功能的名称，登录后台后，在后台管理 -> 菜单添加中添加相应的功能的名称，菜单名称填你想要的名称，比如叫管理首页图片，菜单位置你可以随意选择，一般添加在文章管理里面，你也可以建立自己想要的一级菜单。选好菜单权限，然后填一个菜单路径，这个路径的意思是，当你点下这个新建的功能，会调用哪一个文件加载到 `<frame name="main">` 中，比如 `article/manageIndexPicture`，记住，不要带 `.php`。

提交成功后，刷新页面后，在左边的菜单列表中，你会发现在文章管理里面多了一个管理首页图片。你把鼠标移到上面，可以看到一个路径，`admin.php?action=article/manageIndexPicture` 这个 `GET` 数组，也就是我们刚刚添加的菜单路径，当点击这个菜单的链接时，就会把 `article/manageIndexPicture.php` 加载到 `<frame name="main">` 中。所以，这个时候只要编写我们的 `manageIndexPicture.php` 文件就行了。

这个文件到底放在哪里呢？如果你细心发现，你会发现里面已经拥有的菜单，比如之前用过的菜单添加，当鼠标指向它的时候，你会发现它的地址是 `admin.php?action=admin/menuAdd`，你可以去后台根目录下的 `admin` 文件夹下面看看，会发现一个叫 `menuAdd.php` 的文件，这就说明了，我们添加新功能文件的位置应该放在 `article` 文件夹里面。

如果你需求的一个功能和文章(article)一点关系都没有，而需要一个新的功能模块，比如广告(advertisement)，你可以在后台的根目录里建一个 `advertisement` 目录，然后在里面添加新的功能，比如广告添加(advertisementAdd)，也就是菜单目录填写 `advertisement/advertisementAdd`，然后在 `advertisement` 文件夹里建一个 `php` 文件叫做 `advertisementAdd.php` 即可。

明白了这点之后，只要把你想要的功能写在新建的 `manageIndexPicture.php` 里面就行了。

需要注意的一点是：

在新建的 `php` 文件的开头加上 `require_once(ROOT_PATH."admin/check.php");`
`require_once(ROOT_PATH."inc/inc.php");`
`require_once(ROOT_PATH."inc/table_config.php");`
`check.php` 文件是**必须**的，用来判断用户是否登录，另外两个文件，也是必要的，是一些配置文件，详情可以见文件开头的备注或看下一节 6.文件参考。

6.文件参考

说明：只对关键的文件目录进行了说明，未说明的文件应该都是一些功能模块的文件

`index.php` 入口文件，里面只有一个简简单单的跳转。

`admin.php` 后台的主体文件，在这个文件里判断是否有用户登录，如果没有登录，则会跳转到 `admin/login.php` 文件，如果登录了，则会显示出下面的 `html` 代码，根据默认的 `GET['action']` 值，显示出相应的功能列表与操作页面。

`foot.php` 显示页脚的版权等一些信息，里面的变量在 `inc/conf.php` 中定义。如下图所示。

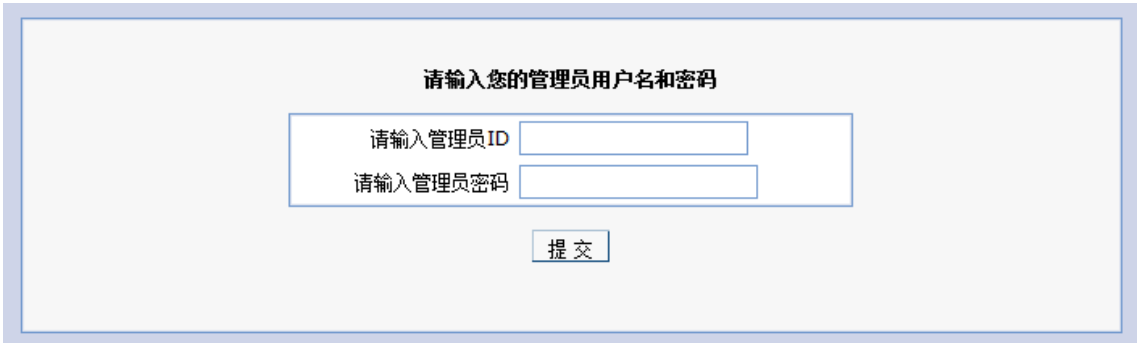


`head.php` 显示页头的三个友情链接的样式，里面的变量也在 `inc/conf.php` 中定义。如下图所示。



`thumb.php` 一个制作缩略图的程序文件。

`admin/login.php` 文件最底部的 `html` 用来显示出登陆框，如下图所示。



文件内的 php 代码部分，都是在判断登录的条件，登录成功后将把当前用户的 id 存在 SESSION 数组中，然后会返回到 admin.php，这时再判断是否登录的时候，就已经通过了这个判断，不会再跳到 admin/login.php 文件了。

[admin/check.php](#) 这个文件是用于判断是否登录过了，在载入绝大数文件的时候（尤其是一些重要的功能），都会先通过这个文件判断一下是否有用户登录，如果登录了，才会继续向下运行，如果没有登录，则会跳到 [admin/quit.php](#) 退出到登录界面。

[admin/quit.php](#) 注销 SESSION 数组，退出登录。

[admin/index.php](#) 这个文件是用来显示后台的系统信息，比如服务器的版本，PHP 的版本，数据库的版本等，是 `<frame name="main">` 默认载入的页面。显示的内容如下图所示。

欢迎光临 网站名称 系统设置面板	
系统说明	
当前版本 V3.2 该后台管理系统供网站名称使用！ 若有任何疑问，请联系 redrock@cqupt.edu.cn	
系统信息	
PHP 程序版本:	5.2.5
MySQL 版本:	5.0.27-community-nt
服务器端信息:	Apache/2.2.22 (Win32) PHP/5.2.5
最大上传限制:	20M
最大执行时间:	30 seconds
服务器所在时间:	2012-11-13 18:49
开发团队	
系统开发: 重庆邮电大学红岩网校	
个人信息	
您上次登录时间:	2012-11-13 18:10:22
您上次登录 IP:	127.0.0.1

[admin/left.php](#) 这个文件是用来显示后台左边的操作列表的，根据当前用户的权限，从数据库的 menu 表中取出取出权限要求低于当前用户的操作，然后输出到 `<frame name="left">` 的框架中。显示的内容如下图所示。

[inc/Administration.php](#) 该文件定义了一个 Admin 类，与管理员登陆的操作有关。

[inc/conf.php](#) 该文件定义了一些类似常量的信息，比如友情链接的名字和地址，以及电话，邮政编码等，主要作用于 head.php 与 foot.php

[inc/function.php](#) 该文件定义了很多自定义的方法，需要的注意的就是 printMessage()这个方法，在所有的操作页面里，执行完某个操作后，所提示的信息，都是由这个方法所输出的内容显示而成。比如，添加一篇新的文章，提交后，会显示：提交成功，返回继续操作。

[inc/inc.php](#) 该文件提供了一个公共的包含，包含了几个重要的公共文件。

[inc/mysql.php](#) 这个文件里定义了一个 mysql 类，这个类主要作用是连接数据库，与所有功能模块中对数据库读写的操作。

[inc/table_config.php](#) 这个文件定义了一些数据库中的表名。

[editor](#) 目录 中是本文编辑器的插件代码。

[upload](#) 目录 主要是保存的是上传的图片，处理后的缩略图与附件

7.数据库参考

说明：部分表及其字段的作用还不太清楚，我在它们的名字前加了*号，若使用者更清楚的了解它们的作用，希望发送邮件告诉我。

表 [admin](#) 存的是管理员账号的信息，包括用户名，密码，上次登录时间等。

[id](#)：唯一主键，每条记录的唯一标识。

[admin_name](#)：用户名，即用户登录时，填写表单那里的登录用户名。

[admin_real_name](#)：用户的真实姓名。

[admin_key](#)：登录时填写的密码，该字符串经过 MD5 加密。

[admin_rights](#)：用户权限，在读取左边操作列表时与某个操作的权限（[menu.rights_level](#)）像比较，判断是否有

这个操作的权限。

`admin_key_remain_times`：用户登录失败后剩余的尝试次数，数值默认为 15，归零后该用户会锁定，登录成功后，将恢复至 15 次。

`admin_locked`：用户是否锁定，1 为锁定，0 为不锁定。锁定的用户登陆后无法操作。

`admin_email`：用户的电子邮箱地址。

`admin_last_loaded_ip`：用户最后一次登录的 IP 地址。

`admin_menu_rights`：用户对哪些操作有特殊权限。

`dep`：该用户属于的部门，存的是 int 型的数值，对应着表 `dep` 中的 `id`。

`admin_last_loaded_time`：用户最后一次登录的时间，存的是 int 型的时间戳，由 PHP 函数 `time()` 取得。

*表 `adminip` 可能存的是限定某些用户不允许登录的 IP。

`id`：唯一主键，每条记录的唯一标识。

`admin_id`：存的是管理员用户账号的 `id` (`admin.id`)。

*`admin_ip`：可能是禁止登录的 IP。

表 `annex` 存的是附件的相关信息。

`ann_id`：唯一主键，每条记录的唯一标识。

`ann_title`：附件的文件名。

*`ann_source`：可能是表示这个附件是附属于那种资源的，比如附属于某篇文章。

`ann_type`：表示该附件属于哪种类型的分类。

`ann_url`：表示该附件的存放位置。

*`ann_show`：可能是附件的备注，或者需要显示出来告诉用户该附件的信息。

`ann_add_user`：存的是添加该附件的用户的 `id` (`admin.id`)。

`ann_add_ip`：表示上传该附件的 IP 地址。

`ann_add_time`：表示上传该附件的时间。

`ann_article_id`：表示该附件与哪一篇文章相关 (`article.art_id`)。

`ann_click`：表示该附件被点击的次数。

`ann_lock`：表示该附件是否被锁定。

表 `article` 存的是文章的信息。

`art_id`：唯一主键，每条记录的唯一标识。

`art_title`：表示该文章的标题。

`art_source`：表示该文章的来源。

`art_author`：表示该文章的作者。

*`art_dept`：可能表示该文章属于哪个部门。

`art_content`：表示该文章的正文内容。

`art_type`：表示该文章所属的类别的 `id` (`type.typeid`)。

`art_add_user`：表示添加该文章用户的 `id` (`admin.id`)。

`art_add_ip`：表示添加该文章的 IP 地址。

`art_add_time`：表示添加该文章时的时间。

`art_click`：表示该文章的点击数量。

`art_locked`：表示该文章是否锁定。

`art_ismv`：表示该文章是否有视频文件。

*`art_dep`：可能表示该文章属于哪个部门。

表 `dep` 存的是部门的信息。

id : 唯一主键, 每条记录的唯一标识。

name : 表示该部门的名字。

表 **execlog** 存的是日常操作的日志。

id : 唯一主键, 每条记录的唯一标识。

user_id : 表示执行该操作用户的 **id** (**admin.id**)。

user_name : 表示执行该操作用户的 **name** (**admin.admin_name**)。

exec_url : 表示该用户执行了哪一条操作。

exec_date : 表示该操作执行时的时间。

exec_ip : 表示该操作执行时的 IP 地址。

表 **menu** 存的是菜单的信息。

id : 唯一主键, 每条记录的唯一标识。

name : 表示该菜单的名字。

url : 表示该菜单链接着哪个文件。

father_id : 表示该菜单隶属于哪一个父菜单下。

order : 菜单的排序。

rights_level : 表示用户能使用这个操作的最小权限, 与用户的权限 (**admin.admin_rights**) 进行比较。

表 **message** 存的是留言的信息。

mid : 唯一主键, 每条记录的唯一标识。

msg_title : 表示该留言的标题。

msg_content : 表示该留言的正文内容。

***father_id** : 可能表示该留言隶属于哪一篇文章。

***is_reply** : 可能表示该留言是否回复的状态。

is_locked : 表示该留言是否锁定。

msg_clicked : 表示该留言被点击的次数。

msg_add_ip : 表示该留言被添加时的 IP 地址。

msg_art_id : 表示该留言与哪一篇文章相关联 (**article.art_id**)。

msg_add_user : 表示该留言被哪一用户所添加 (**admin.id**)。

msg_add_time : 表示该留言被添加的时间。

表 **pic** 存的是图片的相关信息。

pic_id : 唯一主键, 每条记录的唯一标识。

pic_title : 表示该图片的标题/文件名。

pic_source : 表示该图片的来源。

pic_type : 表示该图片隶属于哪一个分类。

pic_url : 表示该图片的原图地址。

pic_thumbnail_url : 表示该图片的缩略图的存放地址。

***pic_show** : 可能表示该图片的备注说明。

pic_add_user : 表示该图片是被哪一用户所添加 (**admin.id**)。

pic_add_ip : 表示该图片被添加时的 IP 地址。

pic_add_time : 表示该图片被添加时的时间。

pic_article_id : 表示该图片与哪一篇文章相关联 (**article.art_id**)。

pic_click : 表示该图片被点击的次数。

pic_locked : 表示该图片是否被锁定。

表 `type` 存的是文章分类的信息。

`typeid` : 唯一主键, 每条记录的唯一标识。

`fatherid` : 二级分类才使用的父分类 id。

`typedetail` : 分类名。

`*type_order` : 可能分类的排序。

`*type_hidden` : 可能表示该分类是否隐藏。

8.附：我的代码规范

说明：仅供参考

8.1.html 代码

标签的内部如果是内容的话, 可以写在一行, 如果里面还有标签的话, 推荐写出层级关系。考虑下面的代码：

```
<div>This is the content</div>
<div>
    <div>This is the content</div>
    .....
</div>
```

没有成对标签的单个标签, 推荐在最后的斜杠前加上一个空格。考虑下面的代码：

```
<hr /> //注意斜杠前面有一个空格
<input type="text" />

<br />
```

如果 html 代码中有一些小样式, 比如居中或者加粗, 不推荐使用一些标签来达到这样的效果, 推荐使用 CSS 来定义样式, 而不使用某些标签的特性来达到某种效果, 考虑下面的代码：

```
<div id="centerTitle">This is a title</div>
//这段文本要居中, 使用 CSS : #centerTitle{text-align:center;}
<center>This is a title</center> //我不推荐这样的写法
<p>My name is <b>Kross</b>.</p> //给字体加粗也不推荐使用<b>标签, 请统一使用 CSS 定义样式
```

8.2.php 代码

变量, 函数的命名, 必须使用正确的英文单词 (组合), 禁止使用汉语拼音与单个字母与数字的组合来组成变量名, 函数名, 并且使用驼峰命名法。考虑下面的代码：

```
$inputUsername, $passwordAfterMD5 //驼峰命名法, 第一个单词小写, 后面的单词第一个字母大写。
$shengri //禁止使用拼音, 推荐使用 $birthday
function createTreeFromArray() //自定义函数名也是如此
```

花括号的打法, 左花括号跟随函数名, 流程, 循环语句的后面, 右花括号换行后与函数名, 流程, 循环语句对齐, 考虑下面的代码：

```
function printMessage(){ //左花括号跟随函数名
    if(.....){
        .....
    }else if(){
```

```

        .....
    }

    while(.....){
        .....
    }    //考虑上述代码的花括号对齐方式

}    //由花括号与函数名那行对齐

```

多参数，或者很多项写在一行，分割号（可能是逗号，也可能是其它符号）后加上空格，考虑下面的代码：

```

printMessage($x, $y, $content, $length)    //逗号后面跟上一个空格，多参数看得更清晰明了
$time == 0? "时间是零": date("Y-m-d H:i:s", $time);
//注意这里的?号与:号后面都跟随了一个空格，date()方法里面的逗号后面也跟随了一个空格

```

关联下标使用单引号，不推荐使用双引号，考虑下面的代码：

```

$_POST['username'];//推荐
$_POST["username"];//不推荐

```

SQL 语句务必要规范，如果有变量，请使用双引号隔开，并用点号连接。考虑下面的代码：

```

$sql = "SELECT `id`, `name`, `age` FROM `student` WHERE `id` = '". $id. "'";
//注意上述 SQL 语句的字段名，表名都用来反撇号 ` 隔开，逗号后面跟随了空格
//SQL 语句中关键字请使用大写，SELECT FROM WHERE 等
//最后的条件语句中，本应该是 `id` = '12' 这种形式，但传入的 id 是一个变量，需要用双引号把 SQL 语句分割开，并且用点号连接

```

运算，比较的语句，请使用空格，考虑下面的代码：

```

$sum = (1 + 2) / 3 //注意等号，加号，除号的左右两边都有空格
if(isset($_POST['action']) && $_POST['action'] == "main"){
//注意&&，==的左右两边都有一个空格
    .....
}

```

9.附：修订日志

2012.11.13 黄劼

首次编写完成。

2012.11.22 黄劼

添加 5.2.更改网站名称的说明，将代码规范的字体由“微软雅黑”改为“Consolas”。

2013.02.21 黄劼

修改了数据库参考