

第 8 章

博客管理系统

(Apache+PHP+phpMyAdmin+MySQL 5.0 实现)

如果有人问“什么是 web2.0? ”，恐怕没几个人能说得清楚，但是要问什么是博客，哪怕是从不上网的人也是耳熟能详的了。博客正是 web2.0 概念中重要的组成部分之一（大家熟知的还包括 IM 即时通和 RSS 阅读器）。

Blog（博客），全名 Weblog，后来缩写为 Blog。Blogger 就是写 Blog 的人，习惯于在网上写出日记、发布个人照片、展示个性自我的用户群体。对于 Blog/Blogger 的中文名称，有翻译成“博客”，也有翻译为“网志”，但大多数人都已经认可了“博客”。

通过阅读本章，读者可以学到：

- » 博客管理系统的开发流程
- » 进一步掌握如何做项目需求分析与系统设计
- » 实现一个简单的公告栏管理模块
- » 掌握不同的图片上传技术
- » 掌握一种采用 JavaScript+CSS 技术来实现的半透明动态下拉菜单

8.1 开发背景

博客管理系统给人们提供抒发个人情感、人与人之间进行良好沟通的平台，博客拥有真实的内容，可以通过博客记录下工作、学习、生活和娱乐的点滴，以及发表文章和评论，从而在网上建立一个完全属于自己的个人天地，成为当今网络最为个性化和平民化的个人展示空间。对于网民，只要拥有博客，就可以超越现实生活，拥有不同以往的全新网上生活；对于博客服务商，则必须从功能提供转型到全方位社会服务的提供，建立虚拟社会，并负责维护运行，保证博客日常生活；对于社会而言，有利于构建和谐互联网空间，维护和谐的社会环境。从这个角度来说，构建新生活方式，将是互联网发展的一个里程碑。

8.2 需求分析

信息时代的今天，博客已经成为一种新的生活方式。在网络中构建一个赋有个性化的个人博客，提供了一种可信任的和实时连通的网络环境，通过网络开放性和交互性的特点，让用户在任何时间、任何地点，通过网络方便地“生活”，不仅是信息传递与获取，还可以进行群体交流和资源共享，展示自我，为个人发展带来新机遇。

通过对多个博客网的调查分析，客户要求本博客管理系统具有以下功能：

- ☒ 要求系统采用 B/S 架构，实现人机交互。
- ☒ 要求系统界面个性化，色彩搭配和谐，很强的视觉冲击力，操作简便。
- ☒ 要求突出主题，显示最新文章和公告。
- ☒ 要求游客可以浏览文章、浏览图片、发表评论。
- ☒ 要求具有强大的搜索查询功能，实现精确查询和模糊查询。
- ☒ 完善的文章管理功能，包括文章的发表、删除，及对文章的评论与回复。
- ☒ 支持图片上传功能，可以上传各种类型的图片。
- ☒ 支持好友功能。
- ☒ 系统运行稳定，安全可靠。

8.3 系统设计

8.3.1 系统目标

该系统主要实现如下目标：

- ☒ 系统采用 B/S 架构，实现人机交互。

- ☑ 系统界面设计以浅色为主, 美观友好, 操作简便。
- ☑ 突出重点内容, 显示最新文章。
- ☑ 非登录用户可以浏览文章、浏览图片、发表评论。
- ☑ 全面的搜索查询功能, 包括精确查询和模糊查询。
- ☑ 完善的文章管理功能, 包括文章的发表、删除, 及对文章的评论与回复。
- ☑ 支持图片上传功能。
- ☑ 支持好友功能。
- ☑ 支持公告栏功能。
- ☑ 系统运行稳定, 安全可靠。

8.3.2 系统功能结构

博客管理系统的功能结构如图 8.1 所示。

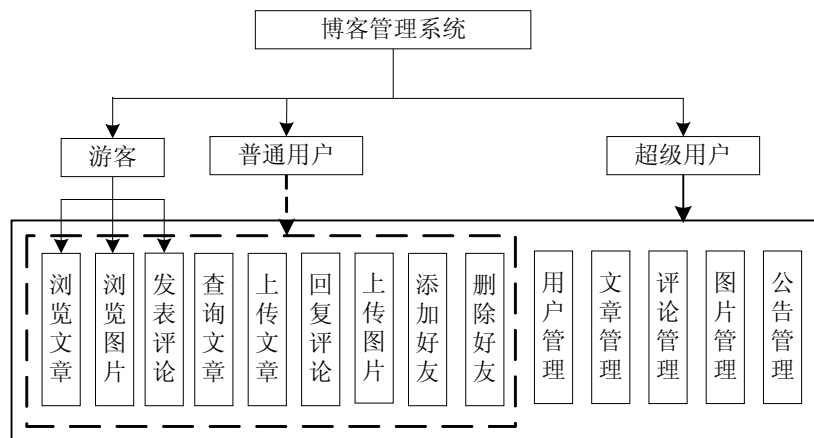


图 8.1 博客管理系统功能结构图

8.3.3 系统功能预览

为了让读者对本系统有个初步的了解和认识, 下面给出本系统的几个页面运行效果图, 如果想查看完整的效果图, 请参见光盘源程序。

博客管理系统的首页如图 8.2 所示, 该页面包含了系统大部分的功能链接, 包括用户注册、用户登录、文章浏览等。用户注册页面如图 8.3 所示, 该页面显示了用户注册时需要填写的资料、注意事项等。

浏览文章页面如图 8.4 所示, 该页面用于显示文章及相关的评论, 也可以发表评论。发表文章页面如图 8.5 所示, 该页面用于登录用户发表文章, 包括文章标题、文字编辑区和文章内容。





图 8.2 博客首页（光盘\TM\08\online\index.php）



图 8.3 用户注册页面（光盘\TM\08\online\Register.php）



图 8.4 浏览文章页面（光盘\TM\08\online\article.php）



图 8.5 发表文章页面（光盘\TM\08\online\file.php）

图片上传页面用于上传图片，如图 8.6 所示，通过该页面用户可以将图片或照片添加到博客中。添加好友页面如图 8.7 所示，该页面用于输入用户好友的详细信息，包括姓名、性别、生日等。



图 8.6 图片上传页面（光盘\TM\08\online\add_pic.php）



图 8.7 添加好友页面（光盘\TM\08\online\friendd.php）

8.3.4 系统流程图

博客管理系统的流程图如图 8.8 所示。



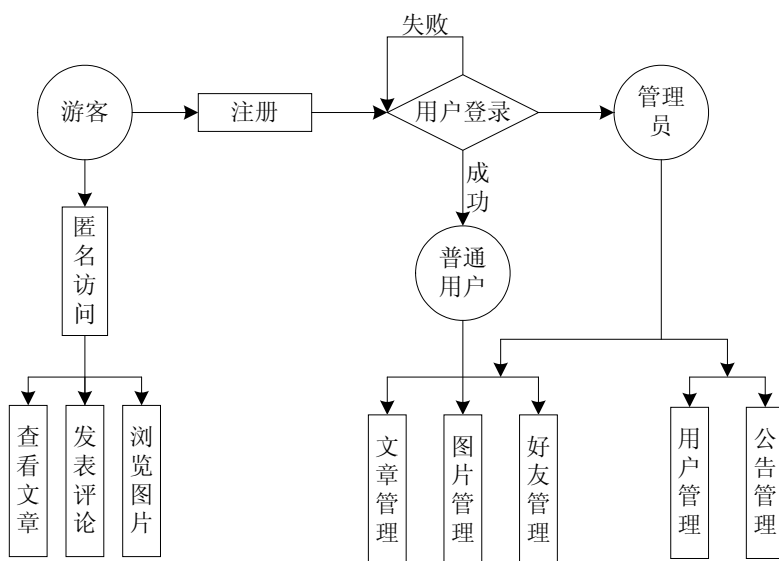


图 8.8 系统流程图

8.3.5 开发环境

在开发博客管理系统平台时，该项目使用的软件开发环境如下：

1. 服务器端

- ☒ 操作系统：Windows Server 2003
- ☒ 服务器：Apache 2.0。
- ☒ PHP 软件：PHP 5.0。
- ☒ 数据库：MySQL 5.0。
- ☒ MySQL 图形化管理软件：PhpMyAdmin-2.5.5。
- ☒ 开发工具：Dreamweaver 8。
- ☒ 浏览器：IE 6.0。
- ☒ 分辨率：最佳效果 1024×768 像素。

2. 客户端

- ☒ 浏览器：推荐使用 IE 6.0 及以上版本。
- ☒ 分辨率：最佳效果 1024×768 像素。

8.3.6 文件夹组织结构

博客系统的目录比较少，结构比较简单，主要有数据库链接文件目录、css 样式目录、js 脚本目录及背景图片目录。文件夹组织结构如图 8.9 所示。



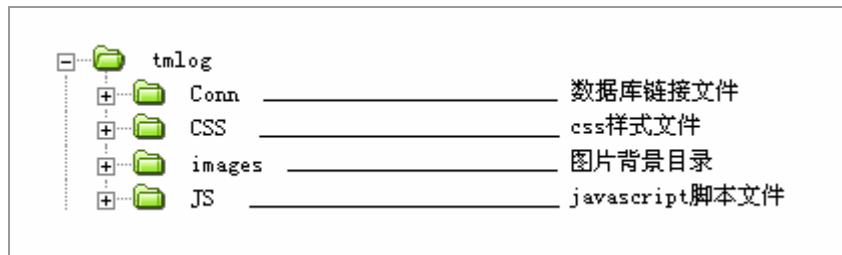


图 8.9 博客管理系统文件夹组织结构

8.4 数据库设计

8.4.1 数据库分析

本系统属于中小型个人网站，毫无争议的，本系统采用的依然是 PHP+MySQL 这对儿黄金组合，无论是从成本、性能、安全上考虑，还是从易操作性上考虑，MySQL 都是最佳选择。

8.4.2 数据库概念设计

通过需求分析和功能上的设计，本系统规划出用户信息实体、上传图片实体、朋友圈实体、文章实体和留言实体。下面给出主要的实体及 E-R 图。

用户信息实体包括注册用户的详细个人信息，如果想在本站中进行发表文章、上传图片等操作，则必须要先进行注。用户信息实体 E-R 图如图 8.10 所示。

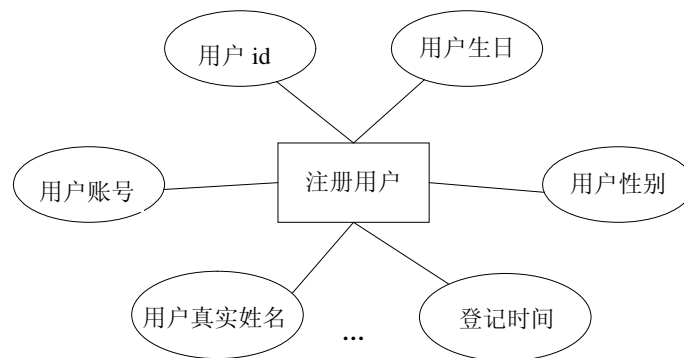


图 8.10 用户信息实体 E-R 图

上传图片实体主要包括上传图片的名称、上传图片用户和上传图片时间等，实体 E-R 图如图 8.11 所示。

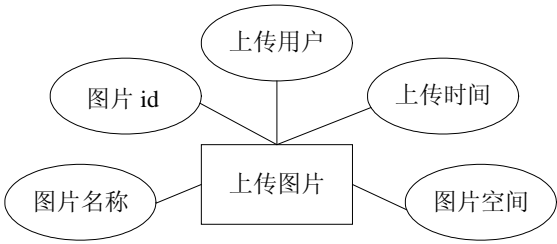


图 8.11 上传图片实体 E-R 图

8.4.3 数据库物理结构设计

根据实体 E-R 图和本系统的实际情况，需要创建 6 张数据表，如图 8.12 所示。

服务器: localhost ▶ 数据库: db_tmlog			
表	类型	整理	说明
tb_article	MyISAM	gb2312_chinese_ci	文章列表
tb_filecomment	MyISAM	gb2312_chinese_ci	评论列表
tb_friend	MyISAM	gb2312_chinese_ci	好友列表
tb_public	MyISAM	gb2312_chinese_ci	公告列表
tb_tpsc	MyISAM	gb2312_chinese_ci	图片列表
tb_user	MyISAM	gb2312_chinese_ci	用户列表

图 8.12 数据表列表

下面来具体看一下这 6 张数据表的结构设计。

1. tb_user（用户列表）

用户列表主要存储用户的个人信息。tb_user 表的结构如图 8.13 所示。

服务器: localhost ▶ 数据库: db_tmlog ▶ 表: tb_user						
字段	类型	整理	属性	Null	默认	说明
id	int(20)			否		自动编号
regname	varchar(20)	gb2312_chinese_ci		否		用户账号
regrealname	varchar(20)	gb2312_chinese_ci		否		真实姓名
regpwd	varchar(40)	gb2312_chinese_ci		否		用户密码
regbirthday	date			否	0000-00-00	用户生日
regemail	varchar(100)	gb2312_chinese_ci		否		E-mail
regcity	varchar(100)	gb2312_chinese_ci		否		所在城市
regico	varchar(50)	gb2312_chinese_ci		否		用户头像
regsex	varchar(4)	gb2312_chinese_ci		否		用户性别
regqq	varchar(40)	gb2312_chinese_ci		否		用户QQ
reghomepage	varchar(100)	gb2312_chinese_ci		否		用户主页
regsign	varchar(200)	gb2312_chinese_ci		否		个人签名
regintroduce	mediumtext	gb2312_chinese_ci		否		自我介绍
ip	varchar(20)	gb2312_chinese_ci		否		登录IP
fig	int(1)			否	0	管理员

图 8.13 用户列表结构

2. tb_article（文章列表）

文章列表存储的是用户发表过的文章信息。tb_article 表的结构如图 8.14 所示。

服务器: localhost ▶ 数据库: db_tmlog ▶ 表 : tb_article						
字段	类型	整理	属性	Null	默认	说明
id	int(10)			否		自动编号
title	varchar(200)	gb2312_chinese_ci		否		文章标题
content	mediumtext	gb2312_chinese_ci		否		文章内容
author	varchar(20)	gb2312_chinese_ci		否		文章作者
now	datetime			否	0000-00-00 00:00:00	上传时间

图 8.14 文章列表结构

3. tb_filecomment（评论列表）

评论列表存储的是用户对文章的评论，包括注册用户和游客都可以发表评论。tb_filecomment 表的结构如图 8.15 所示。

服务器: localhost ▶ 数据库: db_tmlog ▶ 表 : tb_filecomment						
字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
fileid	int(4)			否	0	文章id
username	varchar(20)	gb2312_chinese_ci		否		评论用户
content	text	gb2312_chinese_ci		否		评论内容
datetime	datetime			否	0000-00-00 00:00:00	评论时间

图 8.15 评论列表结构

4. tb_tpssc（图片列表）

图片列表存储的是上传图片的信息，如图片名称、上传用户、上传时间等。tb_tpssc 表的结构如图 8.16 所示。

服务器: localhost ▶ 数据库: db_tmlog ▶ 表 : tb_tpssc						
字段	类型	整理	属性	Null	默认	说明
id	int(10)			否		自动编号
tpmc	varchar(30)	gb2312_chinese_ci		否		图片名称
file	mediumblob		BINARY	否		二进制图片
author	varchar(20)	gb2312_chinese_ci		否		上传用户
scsj	date			否	0000-00-00	上传时间

图 8.16 图片列表结构

5. tb_friend（好友列表）

好友列表主要记录了姓名、性别、生日等好友的个人信息。tb_friend 表的结构如图 8.17 所示。

服务器: localhost ▶ 数据库: db_tmlog ▶ 表: tb_friend						
字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
name	varchar(50)	gb2312_chinese_ci		否		好友姓名
sex	varchar(10)	gb2312_chinese_ci		否		好友性别
bir	date			否	0000-00-00	好友生日
city	varchar(50)	gb2312_chinese_ci		否		所在城市
address	varchar(100)	gb2312_chinese_ci		否		家庭住址
postcode	varchar(6)	gb2312_chinese_ci		否		邮政编码
email	varchar(50)	gb2312_chinese_ci		否		E-mail
tel	varchar(20)	gb2312_chinese_ci		否		座机号码
handset	varchar(20)	gb2312_chinese_ci		否	0	手机号码
QQ	varchar(20)	gb2312_chinese_ci		否	0	好友QQ
username	varchar(20)	gb2312_chinese_ci		否		用户昵称

图 8.17 好友列表结构

6. tb_public（公告列表）

公告列表主要记录了网站情况、博客系统的版本情况或是网站活动等等。公告列表的结构如图 8.18 所示。

服务器: localhost ▶ 数据库: db_tmlog ▶ 表: tb_public						
字段	类型	整理	属性	Null	默认	说明
id	int(4)			否		自动编号
title	varchar(50)	gb2312_chinese_ci		否		公告主题
content	varchar(200)	gb2312_chinese_ci		否		公告内容
pub_time	date			否		发布时间

图 8.18 公告列表结构

8.5 首页设计

8.5.1 首页概述

本系统首页页面设计简洁，主要包括以下 3 部分内容：

- ☑ 首部导航栏：包括首页链接、注册和登录模块。



- ☑ 左侧显示区：包括最新文章、最新图片和系统时间模块。游客主要通过该区域浏览文章、浏览图片及发表评论。
- ☑ 主显示区：为系统公告栏，显示系统及网站的最新咨询。

下面看一下本案例中提供的首页，该首页在本书光盘中的路径为\TM\08\tmlog\index.php，如图 8.19 所示。

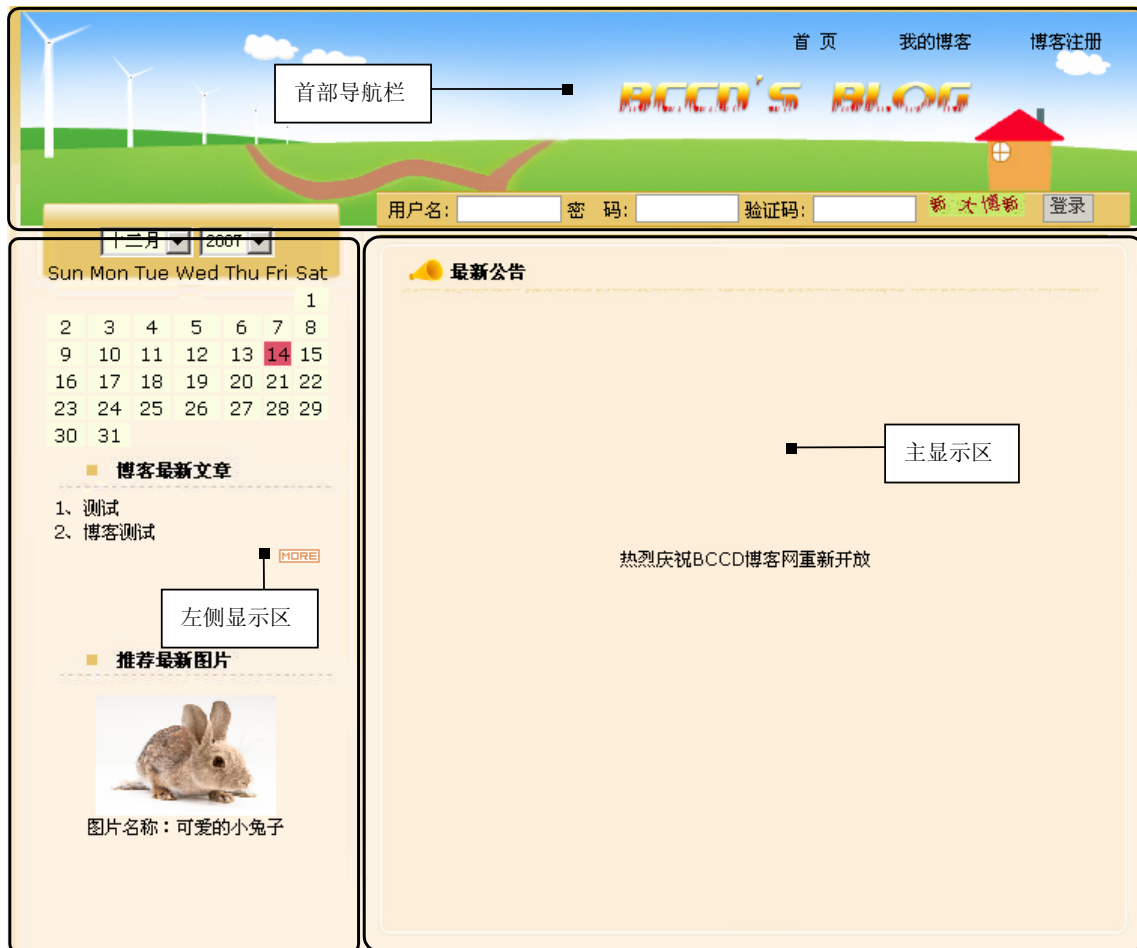


图 8.19 博客管理系统首页

8.5.2 首页技术分析

在首页主显示区，是一个公告栏模块。公告栏主要用于公布系统版本的更新或升级情况、网站的最新活动安排等，也可以链接一些用户的精彩文章。本系统的公告栏模块是通过<marquee>标签来实现的。<marquee>标签是 HTML 自带的、也是初学者最常用的公告栏实现方式。使用<marquee>标签，可以实现文字或图片的滚动效果，增加了实用性的同时，也增加了观赏性。下面就来学习一下这个标签的常用方法及属性。

🔔 注意：<marquee>标签是微软与网景等公司私制的，从来没有被 W3C 当作正式标签来使用，所以除了 IE 浏览器外，有些属性是不被其他浏览器所支持的。在使用时一定要留意。

1. <marquee>标签的文字移动属性

<marquee>标签的特点就是可以是文字或图片动起来，在早些时候，这可是一个了不起的技术。随着 W3C 标准的逐渐完善，<marquee>标签已经越来越少有用武之地了，但有些技术是永远不变的，比如<marquee>标签中属性名称和属性值，了解 css 样式表和 JavaScript 脚本语言的人会感觉到非常熟悉，因为这些属性名称和属性值都经常被提及和使用。<marquee>标签用的文字移动属性及属性值，如表 8.1 所示。

表 8.1 <marquee>标签常用的文字移动属性及说明

属 性 名 称	属 性 值	属 性 说 明	应 用 举 例
marquee	无	标签基本语法，除了文字外，还可以是图片	<marquee>你好，PHP</marquee>
direction	left,right,up,down	文字移动属性，分别表示从右往左、从左往右、从下到上、从上到下	<marquee direction="up">从下到上移动</marquee>
behavior	scroll,slide,alternate	文字移动方式，分别表示沿同一方向不停滚动、只滚动一次、在两个边界内来回滚动	<marquee behavior="scroll">不停的循环播放</marquee>
loop	数值 1,2,3...	循环次数，不指定则表示为无限循环	<marquee loop=5 behavior=slide>只循环滚动 3 次</marquee>
scrollmount	数值 10,20,30...	滚动速度，数值越大，速度越快	<marquee scrollmount=25>我漂起来了！！</marquee>
Scrolldelay	数值 100,200,300...	延时，每动一次，停止的时间	<marquee scrolldelay="500" scrollamount=100>停停走走</marquee>

2. <marquee>标签的外观设置

<marquee>标签不仅可以设置文字移动效果，还能设置标签的外观。外观包括对齐方式、标签底色、面积等。<marquee>标签常用的外观属性及说明如表 8.2 所示。

表 8.2 <marquee>标签常用的外观设置属性及说明

属 性 名 称	属 性 值	属 性 说 明	应 用 举 例
align	top,middle,bottom	指定文本的对齐方式	<marquee align="middle">我在中间</marquee>
bgcolor	颜色值"#000000"	滚动栏的颜色	<marquee bgcolor="#DEEBEF">看到了吗？</marquee>
height	数值 10,20,30...	滚动栏高度	<marquee height="15">不要，太矮了</marquee>
width	数值 10,20,30...	滚动栏宽度	<marquee width="100">哇！好大的房子啊</marquee>

3. <marquee>标签的其他属性

除了以上的常规设置，<marquee>标签还有两个特殊的属性，即 start()和 stop()，它们可以配合 JavaScript 事件产生很有意思的效果。例如：

```
onMouseOut="this.start()";           //当鼠标移出该区域时，开始滚动
onMouseOver="this.stop()";           //当鼠标进入该区域时，停止滚动
```

8.5.3 首页的实现过程

博客管理系统采用二分栏结构，表单布局。具体实现代码如下：

例程01 代码位置：光盘\TM\08\tmllog\index.php

```
<?php
session_start();           //开启session支持
include "Conn/conn.php";   //包含数据库链接文件
?>
<table width="757" border="0" align="center" cellpadding="0" cellspacing="0">
  <tr align="right" valign="top">
    <td height="149" colspan="2" background="images/head.jpg">
      ...                  <!--上部导航栏 -->
    </td>
  </tr>
  <tr>
    <td width="236" height="501" background=" images/left.jpg">
      ...                  <!-- 左侧导航栏-->
    </td>
    <td width="521" height="501" align="center" background=" images/right.jpg">
      <!-- -----系统公告区----- -->
      <?php
        $p_sql = "select * from tb_public order by id desc";           //从数据表中读取数据
        $p_rst = mysql_query($p_sql,$link);                           //执行SQL语句
      ?>
      <!-- -----<marquee>标签开始----- -->
      ❶ <marquee onMouseOver=this.stop() style="WIDTH: 426px; HEIGHT: 280px" onMouseOut=this.start()
scrollamount=2 scrolldelay=7 direction=up>
      <?php
        while($p_row = mysql_fetch_row($p_rst)){                       //循环输出公告标题
          ?>
          ❷ <a href="#" onclick="wopen=open('show_pub.php?id=?php echo
$p_row[0]; ?>',"height=200,width=500,scrollbars=no")"><?php echo $p_row[1]; ?></a><br>
            <?>?>
          </marquee>
        <!-- ----- -->
      </td>
    </tr>
  </table>
```

代码贴士

❶ <marquee>标签：<marquee>标签中包含了 start 和 stop 两个特殊的属性，还有速度属性 scrollamount 和延迟属性 scrolldelay，而公告栏的高度（height）和宽度（width）则是通过 css 样式设置的。

❷ <?php echo \$p_row[1]; ?>：循环输出公告标题，并为每个公告添加超链接，这里用到了 JavaScript 脚本的 open 方法，该方法的作用是打开一个新窗口，同时可以对窗口样式做出设置。Open 方法的一般格式为 open("链接的 url","自定义名称","窗口样式")。

8.6 文章管理模块设计

8.6.1 文章管理模块概述

对一个博客系统来说，文章管理是最基本的功能，但同时也是最复杂的一个功能。本系统的文章管理模块包括“添加博客文章”、“查找博客文章”、“管理我的博客”、“发表评论”“删除文章”和“删除评论”等 6 大功能。其中，普通用户只能删除自己的文章及对文章的评论，只有管理员才有权删除任何一篇文章及回复。下面给出文章管理模块的框架，如图 8.20 所示。

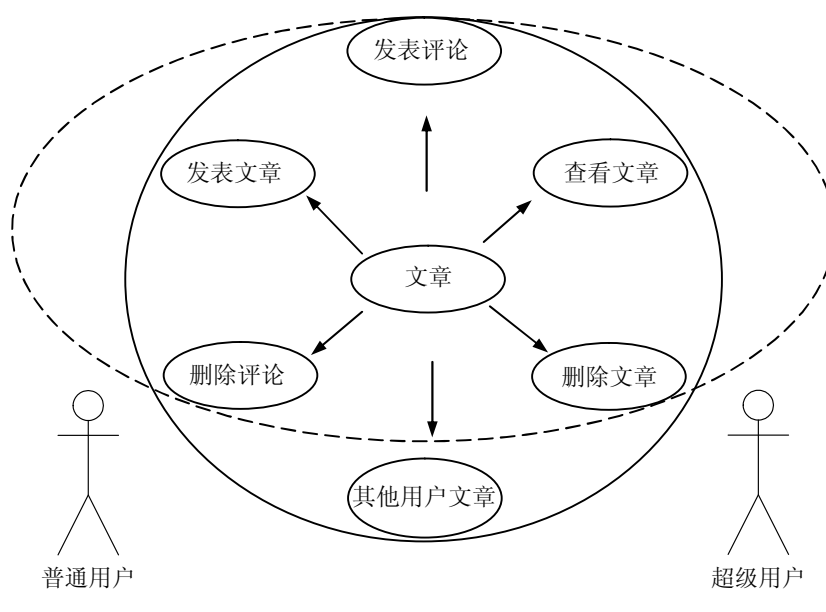


图 8.20 文章管理模块框架图

8.6.2 文章管理模块技术分析

想要使用文章管理模块，前提是用户必须登录，匿名用户是无法访问这些功能的；想要删除文章和评论，前提是当前用户要么是管理员权限，要么是文章拥有者，否则不会显示删除功能的。这两方面的控制都需要 session 的配合，本节就来讲一下 session 的应用及常见的问题处理。

session 的中文译名为“会话”，是指用户从进入网站开始，直到关闭网站这段时间内，所有网页共同使用的公共变量的存贮机制。session 比 cookie 更有优势：session 是存储在服务器端的，不易被伪造；session 的存储没有长度限制；session 的控制更容易等。

PHP 中的 session 功能一直令许多初学者望而生畏，因为经常出现一些莫名其妙的错误，而又不知道如何去解决。其实，大多数的错误都是因为对 session 的配置不了解而使用方法不正确造成的，在 php.ini 中对 session 的配置如表 8.3 所示。



表 8.3 session 的常用配置选项


配 置 选 项	说 明
session.save_path = c:/temp	保存session变量的目录，在linux/unix下为/tmp
session.ues_cookies = 1	是否使用cookie
session.name = PHPSESSID	表示会话ID
session.auto_start = 0	是否自动启用Session，当为1时，在每页中就不必调用Session_sattr()函数了
session.cookie_lifetime = 0	设定cookie送到浏览器后的保存时间，单位为秒。默认值为0，表示直到浏览器关闭
session.cookie_path = /	cookie有效路径
session.cookie_domain =	有效域名
session.serialize_handler = php	定义序列化数据的标识，本功能只有WDDX模块或PHP内部使用，默认值为PHP
session.gc_probability = 1	设定每次临时文件开始处理的处理概率。默认值为1
session.gc_maxlifetime = 1440	设定保存Session的临时文件被清除前的存活秒数
session.referer_check =	决定参照到客户端的Session代码是否要删除。有时出于安全或其他考虑，会设定不删除。默认值为0
session.cache_limiter = nocache	设定Session缓冲限制
session.cache_expire = 180	文档有效期，单位为分钟
session.save_handler = files	用于保存Session变量，默认情况下用文件

对于初学者来说，session 在 php.ini 中是不需要特意去改动，因为安装时会根据操作系统自行做出适当的调整。只在少数的几项，如 session 存活周期（session.cookie_lifetime = 0）、自动开启 session（session.auto_start）等稍加改动即可。

PHP 主要是通过会话（session）处理函数来对 session 进行控制和使用的。常用的处理函数如表 8.4 所示。

表 8.4 PHP 常用的会话处理函数

函 数	函 数 说 明
session_start();	开启session或返回已经存在的session
\$_SESSION['name'] = value;	注册一个session变量
session_id()	设定或取得当前的session_id值
isset(\$_SESSION['name'])	检测指定的session值是否存在。isset不只可以检测session，还可以检测其他类型，如isset(\$_POST['name'])、isset(\$_GET['name'])等
session_regenerate_id()	更改session_id的值
session_name()	返回或改变当前session的name
unset(\$_SESSION['name'])	删除名为name的session。注意
session_destroy()	结束当前会话，删除所有session

 注意：（1）如果要改变当前 session 的 name 值，必须在 session()之前调用 session_name()函数，而且 session_name 不能全部是数字，否则会不停的生成新的 session_id。

（2）不可以写成 unset(\$_SESSION)，这样会禁止整个会话的功能。



1. 尽可能地将 session_start 放到第 1 行

Warning: session_start() [function.session-start]: Cannot send session cache limiter - headers already sent...

2. 在使用 session 之前一定要先写 session start()

```
session_destroy() [function.session-destroy]: Trying to destroy uninitialized session in...
```

3. 删除所有 session

8.6.3 添加文章的实现过程

当用户登录后，系统会直接进入文章添加页（file.php），也可以通过单击“文章管理”/“添加博客文章”回到 file.php 页。文章添加页面的运行结果如图 8.21 所示。



添加文章页为一个发布表单，包括文章主题、文字编辑、文章内容等元素。部分表单元素如表 8.5 所示。

表 8.5 添加文章页面的主要表单元素


名 称	元 素 类 型	重 要 属 性	含 义
myform	form	method="post" action="check_file.php"	添加文章表单
txt_title	text	id="txt_title" size="68"	文章标题
font	select	name="font" class="wenbenkuang" id="font" onChange="showfont(this.options[this.selectedIndex].value)"	文章字体
size	select	class="wenbenkuang" onChange="showsize(this.options[this.selectedIndex].value)"	字体大小
color	select	onChange="showcolor(this.options[this.selectedIndex].value)" name="color" size="1" class="wenbenkuang" id="select"	字体颜色
file	textarea	cols="75" rows="20" id="file" style="border:0px;width:520px;"	文章内容
btn_tj	submit	id="btn_tj" value="提交" onClick="return check();"	“提交”按钮

当用户填写完博客主题和文章内容后，单击“提交”按钮，系统将跳转到处理页（check_file.php）进行处理。在处理页中，将传过来的文章标题、文章作者和文章内容等参数组成 insert 语句，并最终保存到数据表中。如果添加信息成功，系统返回到本页，可继续执行添加操作；如果添加失败，则返回到上一步。程序的关键代码如下：


例程02 代码位置：光盘\TM\08\tmlog\check_file.php

```
<?php
session_start();                //开启session支持
include "Conn/conn.php";        //包含数据库链接文件
if($btn_tj<>""){                //判断传值页面
$title=$_POST[txt_title];      //取得文章标题
$author=$_SESSION[username];   //取得作者
$content=$_POST[file];         //取得文章内容
$now=date("Y-m-d H:i:s");      //使用date函数生成发布时间
/* 生成insert语句 */
$sql="Insert Into tb_article (title,content,author,now) Values ('$title','$content','$author','$now')";
$result=mysql_query($sql);      //执行insert语句
/* 根据$result，返回结果 */
if($result){
    echo "<script>alert('恭喜您，您的文章发表成功!!!');window.location.href='file.php';</script>";
}
else{
    echo "<script>alert('对不起，添加操作失败!!!');history.go(-1);</script>";
}
}
else{
    echo "<script>alert('对不起，添加操作失败!!!');history.go(-1);</script>";
}
}
?>
```



 说明：文章添加页面中，使用了部分 UBB 语法，由于 UBB 语法不属于本书的范畴，所以这里不做讲解，请感兴趣的朋友查看相关的书籍。

8.6.4 文章列表的实现过程

 查看文章列表使用的数据表：tb_article

单击“文章管理”/“我的文章”，将显示用户发表过的文章列表。文章列表页面（myfiles.php）的运行结果如图 8.22 所示。



图 8.22 文章列表页面的运行结果

文章列表页面使用了分页技术和 do...while 循环语句来输出文章标题。程序关键代码如下：

例程03 代码位置：光盘\TM\08\tmlog\myfiles.php

```
<?php
session_start();                                //开启session支持
include "Conn/conn.php";                        //包含数据库链接文件
include "check_login.php";                      //包含权限检查文件
?>
/* 分页 */
<?php
/* $page为当前页，如果$page为空，则初始化为1 */
if ($page==""){
    $page=1;}
❶ if (is_numeric($page)){
    $page_size=20;                                //每页显示20条记录
    $query="select count(*) as total from tb_article where author = '".$_SESSION[username]."' order by id desc";
    $result=mysql_query($query);                  //查询符合条件的记录总条数
    $message_count=mysql_result($result,0,"total"); //要显示的总记录数
    $page_count=ceil($message_count/$page_size);  //根据记录总数除以每页显示的记录数求出所分的页数
    $offset=($page-1)*$page_size;                //计算下一页从第几条数据开始循环
    $sql=mysql_query("select id,title from tb_article where author = '".$_SESSION[username]."' order by id desc limit
    $offset, $page_size");
    $info=mysql_fetch_array($sql);
    ?>
    ...
<?php
```



❶ `is_numeric($page)`: `is_numeric()`函数判断变量`$page`是否为数字，如果是数字，则返回 `true`，否则返回 `false`。如果一组字符全部由数字组成，那么 `is_numeric()`函数也将返回 `true`。

• 390 •

8.6.5 查看文章、评论的实现过程

查看文章、评论使用的数据表：tb_article、tb_filecomment

单击列表中任意一个文章标题，都会看到对应的文章内容和文章评论。查看文章页面(showmy.php)的运行结果如图 8.23 所示。



图 8.23 查看文章页面的运行结果

系统根据当前页面传过来的文章 id 值从数据表 tb_article 中返回对应的文章信息（包括文章 id、文章作者、文章标题、文章内容和发表时间）、输出文章信息后，开始查找表 tb_filecomment 中 fileid 字段值等于文章 id 的所有评论集，并通过分页显示出来。显示文章页面（showmy.php）的关键代码如下：

例程04 代码位置：光盘\TM\08\tml\showmy.php

```
<?php
session_start();
include "Conn/conn.php";
include "check_login.php";
$file_id1=$_GET[file_id];
$bool = false;

//开启session支持
//包含数据库链接文件
//包含权限检查文件
//取得文章ID号
//判断用户删除权限，默认false是没有权限
```

```

?>
    <!-- 首部导航栏 -->
...
<!-- ----- -->
    <!-- 文章显示区 -->
<?php
    $sql=mysql_query("select * from tb_article where id = ".$file_id1); //根据文章ID号查找文章
    $result=mysql_fetch_array($sql); //返回结果集
?>
<!-- -----显示文章----- -->
<table width="100%" border="1" cellpadding="1" cellspacing="1" bordercolor="#D6E7A5" bgcolor="#FFFFFF"
class="i_table">
    <tr bgcolor="#FFFFFF">
        <td width="14%" align="center">博客ID号</td>
        <!-- -----显示文章id----- -->
        <td width="15%"><?php echo $result[id]; ?></td>
        <td width="11%" align="center">作者</td>
        <!-- -----显示文章作者----- -->
        <td width="18%"><?php echo $result[author]; ?></td>
        <td width="12%" align="center">发表时间</td>
        <!-- -----显示发表时间----- -->
        <td width="30%"><?php echo $result[now]; ?></td>
    </tr> <tr bgcolor="#FFFFFF">
        <td align="center">博客主题</td>
        <!-- -----文章主题----- -->
        <td colspan="5">&nbsp;&nbsp;&nbsp;&nbsp;<?php echo $result[title]; ?></td>
    </tr> <tr bgcolor="#FFFFFF">
        <td align="center">文章内容</td>
        <!-- -----文章内容----- -->
        <td colspan="4"><?php echo $result[content]; ?></td>
    </tr>
</tr>
<?php
    /* 判断登录用户是否为管理员，或者是文章的作者 */
    if($_SESSION[fig]==1 or ($_SESSION[username] == $result[author])){
    /* 如果是，就将$bool设为true */
        $bool = true;
    }

?>
    <!-- -----同时显示“删除”按钮，并将文章id作为url后缀一起传到处理页----- -->
    <a href="del_file.php?file_id=<?php echo $result[id];?>"></a>
</tr>
</tr>
</tr>
...
//分页代码部分略
$sql=mysql_query("select * from tb_filecomment where fileid='$file_id1' order by id desc limit $offset, $page_size");
$result=mysql_fetch_array($sql);
/* *****根据返回记录集输出记录***** */
if($result==false){
    echo "<font color=#ff0000>对不起，没有相关评论!</font>";
} else {
    do {
?>
        <tr>
            <td height="57" align="center" valign="top"><table width="480" border="1" cellpadding="1"


```

```

cellspacing="1" bordercolor="#D6E7A5" bgcolor="#FFFFFF" class="i_table">
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|
        <!-- -----评论id----- -->
        <?php echo $result[id]; ?></td>
 <!-- -----评论人----- -->         <?php echo $result[username]; ?></td>  <!-- -----评论时间----- -->         <?php echo $result[datetime]; ?></td>  |         <!-- -----评论内容----- -->         <?php echo $result[content]; ?></td>    |  | | | | | | |

```

8.6.6 删除文章、评论的实现过程

 查看文章、评论模块使用的数据表：tb_article、tb_filecomment

在查看文章评论页面，当系统判定当前用户为管理员或是文章作者时，在每篇文章和评论的后面，都将显示相应的“删除”按钮。单击任意的“删除”按钮，系统会提示是否删除，如果确认，将跳转到处理页（del_file.php 和 del_comment.php），完成删除操作。

在删除文章的处理页中，删除文章的同时，也删除了该篇文章的相关的评论。处理页首先在文章列表（tb_article）中删除 id 等于 \$file_id 的记录，如果没有可删除记录，则提示失败，并返回上一步；如果删除成功，则转到评论列表（tb_filecomment）中，删除所有该篇文章的评论。删除文章页（del_file.php）的关键代码如下：

例程05 代码位置：光盘\TM\08\tml\del_file.php

```

<?php
    session_start();                //开启session支持
    include "check_login.php";       //包含权限检查文件
    include "Conn/conn.php";         //包含数据库链接文件

```



```

$sql="delete from tb_article where id=".$file_id;           //删除文章的delete语句
$result=mysql_query($sql,$link);                          //执行删除操作
if($result){
    $sql1 = "delete from tb_filecomment where fileid = ".$file_id; //删除相对应的文章评论
    $rst1 = mysql_query($sql1,$link);                       //执行删除评论的操作
    if($rst1)
        echo "<script>alert('博客文章已被删除!');location='myfiles.php';</script>";
    else
        echo "<script>alert('删除失败!');history.go(-1);</script>";
}
else{
    echo "<script>alert('博客文章删除操作失败!');history.go(-1);</script>";
}
?>

```

因为删除文章的过程也包含了删除评论的过程，所以这里就不给出删除评论的代码了。

8.7 图片上传模块设计

由于动态网络编程技术的诞生，使得网络更加人性化，为了能够和用户更好地互动，很多网站都提供了让用户上传图片的功能。一个网站拥有图片上传功能是非常必要的。

8.7.1 图片上传模块概述

图片上传在动态网页开发过程中应用非常广泛。如果有比较好的图片想和其他人一同分享，就可以通过图片上传功能来实现，以增加网站的核心竞争力。本系统的图片上传模块主要实现对图片的添加、浏览、查询和删除操作，而对图片的删除则只有管理员才有权限。图片上传模块框架图如图 8.24 所示。

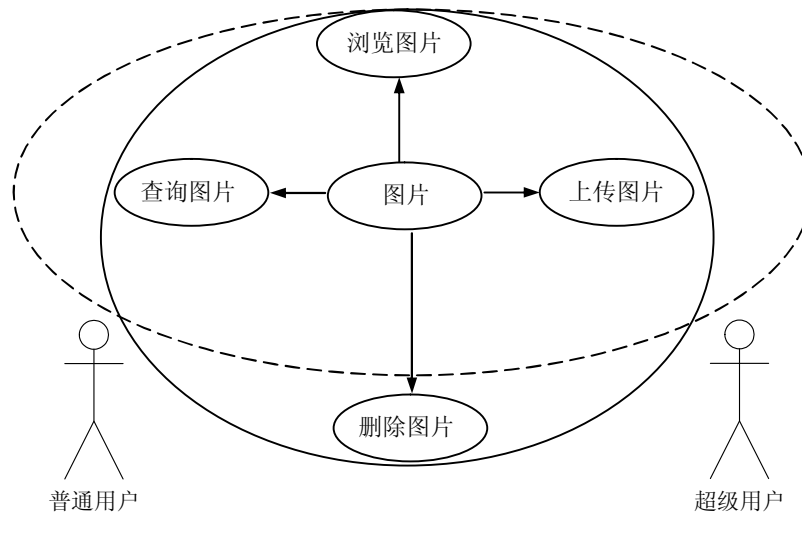


图 8.24 图片上传模块框架图

8.7.2 图片上传模块技术分析

既然是图片上传模块，显而易见本节的主要技术就是上传图片功能了。上传图片 and 上传文件的原理基本相同，下面就来一步一步地学习如何上传图片 and 图片的两种保存方式。

1. 上传图片的基本流程

在网页中实现上传图片功能的步骤如下：

(1) 通过<form>表单中的 file 元素选取上传数据。

使用 file 元素上传数据时注意一点：就是在 form 表单中要加上属性 enctype="multipart/form-data"，否则上传不了文件（图片）。

(2) 在处理页中使用\$_FILES 变量中的属性判断上传文件类型和上传文件（图片）大小是否符合要求。

\$_FILES 变量为系统预定义变量，保存的是上传文件（图片）的相关属性。使用格式为：

```
$_FILES[name][property];
```

图片的相关属性如表 8.6 所示。

表 8.6 \$_FILES 的相关属性

属 性 值	说 明
name	上传文件的文件名
type	上传文件的类型
size	上传文件的大小
tmp_name	上传文件在服务器中的临时文件名
error	上传文件失败的错误代码

(3) 使用 move_uploaded_file() 函数上传文件（图片）或将文件（图片）以二进制的形式保存到数据库中。

使用函数将文件（图片）保存到对应的文件夹中和以二进制的形式保存到数据库中是上传文件（图片）的两种形式，稍后将单独做介绍。

(4) 返回页面等待下一步操作。

2. 使用上传函数保存文件（图片）

使用上传函数上传文件（图片）的本质就是将文件（图片）从浏览器端复制到服务器端指定的文件夹里，数据库所存储的就是文件（图片）的相对地址。当页面显示图片时，实际是分两步：第一步是读取数据表中的地址；第二步是根据地址找到并在页面中显示图片。使用目录保存文件的好处是减少了数据库的容量和对数据库的压力，而且图片很容易被搜索引擎抓到，从而提高网站流量和人气。

move_uploaded_file()函数的格式为：

```
bool move_uploaded_file ( string filename, string destination );
```

☑ filename：上传到服务器中的临时文件名。

☑ destination: 保存文件的实际路径。

🔔 注意: 这里的 filename 为临时文件名, 而不是上传文件的原文件名, 可以通过\$_FILES[filename][tmp_name]来获取。

下面看一个实例, 程序代码如下:

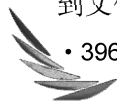
```
<label>请选择要上传的图片（文件小于500K）： </label>
<!--上传文件的form表单-->
<form method="post" action="#" enctype="multipart/form-data">
    <input type="hidden" name="action" value="upload" />
    <input type="file" name="u_file"/>
    <input type="submit" value="上传" />
</form>
<!------->
<?php
    /* 判断是否为上传动作 */
    if(isset($_POST[action])){
        $file_path = "./"; //上传文件存放路径, ./为当前目录下
        $profix = array(".jpg", ".gif", ".jpeg"); //设置允许上传的文件后缀类型
        $f_name = $_FILES[u_file][name]; //取得要上传的文件名
        $pro_name=substr($f_name, strrpos($f_name, ".")); //取得上传文件的后缀
        if(!in_array($pro_name, $profix)){ //判断上传文件的类型是否为允许类型
            echo "文件格式不对";
            exit();
        }
        if($_FILES[u_file][size] <= 0){ //判断上传文件的大小
            echo "文件上传错误, 或文件大于500K,请重新上传"; //如果文件过大, 提示错误
        }
        else{
            /* 上传文件函数 */
            move_uploaded_file($_FILES[u_file][tmp_name], $file_path.$_FILES[u_file][name]);
            echo "上传成功!";
        }
    }
?>
```

3. 使用二进制保存图片

上传图片的另一种保存方式是以二进制的形式存储在数据库中。在计算机看来, 再美的图片、再感人的文章, 也不过就是一堆“0111010011...”的代码段而已, 和其他数据没有什么区别, 存到数据表中都是一样的。存储图片不再需要使用 move_uploaded_file() 函数, 在处理页中直接使用二进制的形式读取出文件图片, 并存到数据表中。本系统采用的就是二进制的形式来保存图片, 具体操作将在实现过程里给出。

使用二进制来保存图片的好处是安全, 特别是涉及到个人隐私, 不易被窃取; 存储方便, 和表中其他数据同等操作就可以; 节省磁盘空间, 由于文件系统类型的限制, 放到数据表中的数据一定比直接放到磁盘中要节省空间。

当然二进制形式的图片也有很大弊端, 在传出的过程中, 如果某个二进制位出现丢失或损坏, 那么整张图片将无法显示, 而是变为一堆乱码, 而且如果存储的图片过多, 造成数据库过于庞大, 影响其他数据的正常读取, 增大服务器的负担。所以, 对于太大的文件, 就不适宜保存到数据库中而是放到文件夹。



8.7.3 图片上传的实现过程

查看文章、评论模块使用的数据表：tb_tpse

博客用户登录后，单击导航栏中的“图片管理”/“添加图片”选项，即可进入添加图片页面，在“图片名称”文本框中添加上传的图片名称，在“上传路径”文本框中选择或者单击“浏览”按钮选择自己喜欢的图片，单击“提交”按钮，以二进制形式将图片上传到数据库中。图片上传页面的运行结果如图 8.25 所示。



图 8.25 图片上传页面的运行结果

图片上传页是一个上传文件的表单，主要包括一个文本域、一个文件域和一个“提交”按钮。部分表单元素的名称及属性如表 8.7 所示。

表 8.7 图片上传页面中的表单元素

名 称	元 素 类 型	重 要 属 性	含 义
myform	form	method="post" action="tptj_ok.php" enctype="multipart/form-data"	图片上传表单
tpmc	text	type="text" id="tpmc" size="40"	图片名称
file	file	type="file" size="23" maxlength="60"	上传路径
btn_tj	submit	type="submit" id="btn_tj" value="提交" onClick="return pic_chk();"	“提交”按钮

当用户输入图片名称，并选择图片路径后，单击“提交”按钮，系统将进入到上传处理页(tptj_ok.php)中进行处理。在处理页中，首先对图片名称进行处理，去掉特殊字符、空行和空格，然后对上传的文件进行类型检查、文件大小检查。最后以二进制的形式，和图片的其他信息（如上传用户、上传时间等）一起存进数据表中。关键代码如下：

例程06 代码位置：光盘\TM\08\tmlog\tptj_ok.php

```
<?php
```

```

session_start(); //开启session支持
include "check_login.php"; //包含权限检查文件
include "Conn/conn.php"; //包含数据库链接文件
if($_POST["btn_tj"]=="提交"){
    ❶ $tpmc=htmlspecialchars($tpmc); //将图片名称中的特殊字符转换成HTML格式
    ❷ $tpmc=str_replace("\n","<br>",$tpmc); //将图片名称中的回车符以自动换行符取代
    $tpmc=str_replace(" ","&nbsp;",$tpmc); //将图片名称中的空格以"&nbsp;"取代
    $author=$_SESSION[username];
    $scsj=date("Y-m-d"); //设置图片的上传时间
    $profix = array(".jpg",".gif",".jpeg"); //设置允许上传的文件后缀类型
    $f_name = $_FILES[file][name]; //取得要上传的文件名
    $pro_name=substr($f_name,strrpos($f_name,". ")); //取得上传文件的后缀
    /* 判断上传文件的类型是否为允许类型 */
    ❸ if(!in_array(strtolower($pro_name), $profix)){
        echo "<script>alert('文件格式不对');history.go(-1);</script>";
        exit();
    }
    /* 判断上传文件的大小, 如果文件过大, 提示错误 */
    if($_FILES[file][size] > 500000){
        echo "<script>alert('文件上传错误,请重新上传');history.go(-1)</script>";
        exit();
    }
    else{
        $fp=fopen($file,"r"); //以只读方式打开文件
        $file=addslashes(fread($fp,filesize($file))); //将文件中的引号部分加上反斜线
        $query="insert into tb_tpse (tpmc,file,author,scsj) values ('$tpmc','$file','$author','$scsj')";
        $result=mysql_query($query);
        echo "<meta http-equiv='refresh' content='1;url=browse_pic.php'>图片上传成功, 请稍等...";
    }
}
?>

```

代码贴士


- ❶ htmlspecialchars()函数: 将特殊字符转换成 HTML 格式。
- ❷ str_replace()函数: 取代所有在字串中出现的字串。语法如下:

```
mixed str_replace ( mixed search, mixed replace, mixed subject , int &count )
```

str_replace()函数将所有在参数 subject 中出现的 search 以参数 replace 替换, 参数&count 表示替换字符串执行的次数。

- ❸ strtolower: 将字符转换为小写字母。

8.7.4 图片浏览的实现过程

 图片上传使用的数据表: tb_tpse

无论是注册用户, 还是非注册用户, 只要登录网站, 就可以无条件地浏览所有图片。而删除图片除了管理员, 其他人都无权操作。非注册用户可以通过首页中的“最新图片”进入图片浏览页面, 注册用户先进入个人管理界面, 单击“图片管理”/“浏览图片”菜单, 同样可以进入图片浏览页面。注

册用户浏览图片页面的运行结果，如图 8.26 所示。



图 8.26 浏览图片页面的运行结果

本页的实现代码和查看文章页面略有不同，查看文章页面中，每条数据占了一行，而查看图片则采用的是分栏显示，以每行两张图片的格式输出，每页显示 4 张图片。通过单击小图片，可以查看图片原效果图。程序关键代码如下：

例程07 代码位置：光盘\TM\08\tmlog\browse_pic.php

```
<?php
    session_start();                                //开启session支持
    include "Conn/conn.php";                        //包含数据库链接文件
    include "check_login.php";                      //包含权限检查文件
?>
...
<!-- 首部导航栏 -->
<tr align="left" colspan="2" >
    <td width="390" height="25" colspan="3" valign="top" bgcolor="#EFF7DE"> <span class="tableBorder_LTR">浏览图片</span></td></tr>
<tr>
    <td height="192" align="center" valign="top" >
        <?php
            ...
            //分页显示
            $query="select * from tb_tpse where scsj order by id desc limit $offset, $page_size";
            /* 返回结果集 */
            $result=mysql_query($query);
        ?>
        <table width="496" border="1" align="center" cellpadding="3" cellspacing="1" bordercolor="#D6D7D6">
```

```

<tr>
<?php
/* 设置一个变量,判断当前数据是奇数输出还是偶数输出 */
    $i=1;
    while($info=mysql_fetch_array($result))
    {
/* 如果当前为偶数输出,记录输出完毕后结束<tr>标签 */
        if($i%2==0){
            ?>
            <td width="500"><table width="245" border="0" cellpadding="0" cellspacing="0">
                <tr>
                    <td colspan="2"><div align="center">
                        <!--显示图片页image.php-->
                        <a href="image.php?recid=<?php echo $info[id]; ?>" target="_blank"></a></div></td></tr>
                        <tr>
                            <td width="109" height="25" align="left">&nbsp;图片名称:<?php echo
$info[tpmc]; ?></td>

                            <td width="128">上传时间:<?php echo $info[scsj]; ?></td></tr>
                        <tr>
                            <td colspan="2" height="25">
                                <?php
                                /* 如果登录用户为管理员,显示删除操作 */
                                if ($_SESSION[fig]==1){
                                    ?>
                                    <a href="remove.php?pic_id=<?php echo $info[id]; ?>"></a>
                                    <?php
                                    }
                                    ?>
                                </td>
                            </tr>
                        </table></td>
                    </tr>
                <?php
            }
/* 否者,开始新的一行 */
            else
            {
                ?>
                <td width="500"><table width="236" border="0" cellpadding="0" cellspacing="0">
                    <tr>
                        <td colspan="2"><div align="center"><a href="image.php?recid=<?php echo $info[id]; ?>"
target="_blank"></a></div></td>
                    </tr>
                    <tr>
                        <td width="110" height="25">&nbsp;图片名称:<?php echo $info[tpmc]; ?></td>
                        <td width="126">上传时间:&nbsp;<?php echo $info[scsj]; ?></td>
                    </tr>
                </table>
            }
        }
    }
}

```

```

        <tr>
            <td colspan="2" height="25">
                <?php
                    if ($_SESSION[fig]==1){
                        ?>
                        <a href="remove.php?pic_id=<?php echo $info[id]?>"></a>
                        <?php
                            }
                        ?>
                    }
                </td>
            </tr>
        </table></td>
    <?php
        }
        /* 变量$i自+1, while循环结束 */
        $i++;
    }
    ?>
</tr>

```


通过上述代码可以看到，预览图片和显示图片实际调用的都是同一页面：image.php。image.php 页就是用来显示图片的实际效果。实现代码如下：

例程08 代码位置：光盘\TM\08\tmlog\image.php


```

<?php
    include "Conn/conn.php";                //包含数据库链接文件
    $query="select id,tpmc,file,author,scsj from tb_tpse where id=".$srecid;           //根据id生成查询语句
    $result=mysql_query($query);             //执行查询语句
    if(!$result) die("error: mysql query");   //判断是否有返回结果
    $num=mysql_num_rows($result);            //计算结果集个数
    if($num<1) die("error: no this recorder");
    $data = mysql_result($result,0,"file");   //返回图片数据
    echo $data;                              //输出图片
?>

```

 **注意：**mysql_result()函数不能和其他处理结果集函数混合调用。

8.7.5 删除图片的实现过程

 删除图片使用的数据表：tb_tpse

删除图片是管理员才有的权限，在图片浏览的实现代码中，已经给出如何判断登录用户是否为管理员。当管理员单击“删除”超链接时，处理页（remove.php）会根据传过来的 id，删除对应的数据表中的数据。实现代码如下：

例程09 代码位置：光盘\TM\08\tmlog\remove.php

```

<?php

```



```

session_start();                                //开启session支持
include "check_login.php";                      //包含权限检查文件
include "Conn/conn.php";                       //包含数据库链接文件
if($_SESSION[fig] != "1"){                     //判断是否为管理员
    echo "<script>alert('您没有删除权限 ');history.go(-1);</script>";
}
$sql="delete from tb_tpse where id=".$_SESSION[id]; //生成删除语句
$result=mysql_query($sql);                     //执行删除语句
if($result){                                   //根据$result返回结果
    echo "<script>alert('图片删除成功!');location='browse_pic.php';</script>";
}
else{
    echo "<script>alert('图片删除操作失败!');history.go(-1);</script>";
}
?>

```

8.7.6 单元测试

在进行上传模块的测试时发现：当上传比较大的图片时，文件大小的验证部分并没有生效，而是直接跳过。在处理页中产生错误，并且显示页面不显示图片。运行结果如图 8.27 所示。

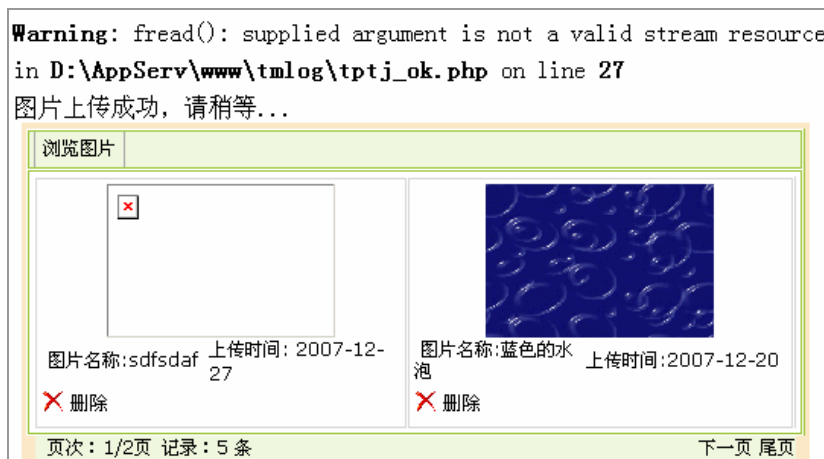


图 8.27 上传图片错误信息及显示结果

查看原文件大小验证的代码如下：

```

/* 判断上传文件的大小，如果文件过大，提示错误 */
if($_FILES[file][size] > 500000){
    echo "<script>alert('文件上传错误，请重新上传');history.go(-1)</script>";
    exit();
}

```

经反复测试后发现：如果上传图片大于 500KB 而小于 2MB 时，原代码生效。如果图片大于 2MB，则代码被跳过，这时输出 `$_FILES[file][size]` 的值为 0。原来，在配置文件 `php.ini` 中，`upload_max_filesize=2MB`，也就是说，上传文件的大小超过了系统设置的值，那么 `$_FILES[file][size]`

返回 0，在判断表达式中返回 **true**，代码自然被跳过。

改写验证代码，更改后的代码如下：

例程10 代码位置：光盘\TM\08\tmllog\tpjtj_ok.php

```
if(($_FILES[file][size] <= 0) or ($_FILES[file][size] > 500000)){
    echo "<script>alert('文件上传错误，或文件大于500K，请重新上传');history.go(-1)</script>";
    exit();
}
```

进行二次测试，问题没有再出现，系统正常运行。

8.8 朋友圈模块设计

8.8.1 朋友圈模块概述

本系统的朋友圈模块的主要功能是添加、查询、删除好友，添加的好友除了该用户以外，包括管理员在内的所有外人都不可以查看，以保证其个人隐私不被外泄。用户被删除时，该用户现有的朋友圈也一并被删除。朋友圈模块框架图如图 8.28 所示。

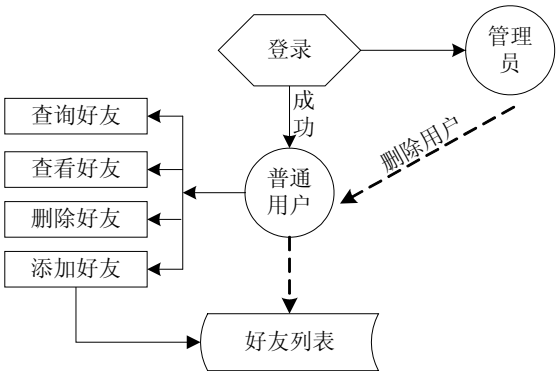


图 8.28 朋友圈模块框架图

8.8.2 朋友圈模块技术分析

在查询好友的功能中，使用到了模糊查询语句，用于模糊查找好友列表。模糊查询语句使用的是 **like** 运算符。在 PHP 中，带有 **like** 运算符的查询语句的常用格式有 2 种：

（1）使用通配符“%”的 **where** 子句

通配符“%”表示 0 个或多个、任意长度和类型的字符，包括中文汉字。

示例 1：表示查找所有内容包含“好”字的文章。

```
select * from tb_file where content like '%好%';
```

示例 2：查找所有包含“好”字或“高”字的文章，这时可以配合 **or** 运算符来使用。代码如下：




```
select * from tb_file where content like '%好%' or content like '%高%';
```

(2) 使用通配符 “_” 的 where 子句

通配符 “_” 表示匹配任意的单个字符。


示例 1：查找用户名只包含 5 个字符，其中后 4 个字符为 soft 的用户。代码如下：

```
select * from tb_user where regname like '_soft';
```


示例 2：查找所有以 t 开头、并且以 t 结尾的、中间包含 3 个字符的用户。代码如下：

```
select * from tb_user where regname like 't___t';
```

查找的结果为：tsoft。

 **注意：**使用 MySQL 做模糊查询要注意编码问题。如果编码不统一，那么查询时就容易查不到数据，或返回的数据不匹配。所以在安装 MySQL 时，要保持和系统编码的统一。常用的编码格式有 gb2312、ISO-8859-1、utf8 和 gbk 等。

8.8.3 查询好友的实现过程

 查看文章、评论模块使用的数据表：tb_friend

当用户要查询好友时，单击“朋友圈管理”/“查询朋友信息”，显示查询页面。查询可以分为姓名查询和编号查询，均为模糊查询。当用户输入要查找的关键字后，单击“检索”按钮，或按 Enter 键，系统跳到处理页进行处理。查询好友页面的运行结果如图 8.29 所示。



查询结果					
编号	28	姓名	tsoft	性别	0
生日	1980-01-14				
所在城市	上海崇明		家庭住址	吉林	
邮政编码	13****		家庭电话	0431-12345***	
e-mail	zh#####@163.com				
手机号码	1351XXXX359		QQ号	1234****	
				 删除	

图 8.29 查询好友页面的运行结果

查询页包含一个查询表单，包括查询条件和查询关键字两部分表单元素。主要表单元素如表 8.8 所示。

表 8.8 查询页表单的主要元素属性

名 称	元 素 类 型	重 要 属 性	含 义
myform	form	method="post" action="query_friend.php" onSubmit="return check();"	查询好友表单
sel_tj	select	<option value="name" selected>姓名</option> <option value="id">编号</option>	查询条件选择
sel_key	text	id="sel_key" size="30"	查询关键字
submit	submit	type="submit" name="submit" value="检索"	“检索”按钮

当处理页接收到查询条件及查询关键字后，生成模糊查询语句，执行 SQL 语句并返回查询结果。如果没有输入关键字，则弹出提示框；如果没有查找到任何结果，则输出“Sorry!没有您要找的朋友!”。处理页的关键代码如下：

例程 11 代码位置：光盘\TM\08\tmlog\query_friend.php

```
<?php
session_start();                                //开启session支持
include "Conn/conn.php";                        //包含数据库链接文件
include "check_login.php";                      //包含权限检查文件
?>
<script language="javascript">
/*  JavaScript脚本，验证输入信息，如果为空，则弹出提示框  */
function check(form){
    if (document.myform.sel_key.value==""){
        alert("请输入查询条件!");
        myform.sel_key.focus();
        return false;
    }
}
</script>
...
<?php
    if ($_POST[sel_key]!=""){
        $tj=$_POST[sel_tj];                    //接收查询字段
        $key=$_POST[sel_key];                  //接收查询关键字
        $sql=mysql_query("select * from tb_friend where $tj like '%$key%' and username='$_SESSION[username]'");
        $result=mysql_fetch_array($sql);        //执行查询语句
        if($result==false){
            echo ("<font color=red>Sorry!没有您要找的朋友!</font>");
        }else{
            <!-- 显示记录表单 -->
        }
    }
?>
...
<?php }?>
```

8.9 开发技巧与难点分析

想象一下：动态的下拉菜单、半透明的背景效果，鼠标划过时的色彩交替，给人一种亦真亦幻的

视觉冲击和美的感受，这样的网站具有更大的吸引力。本系统个人管理页面中的管理菜单实现的就是这种效果。当用户登录到博客网站后，将鼠标移动到“朋友圈管理”等导航链接上时，将在下方显示出半透明的下拉式菜单，透过此下拉菜单仍可以看到页面上的内容，运行效果如图 8.30 所示。



图 8.30 动态半透明效果的下拉菜单

实现半透明背景的下拉菜单，首先需要在页面中实现下拉菜单，然后再通过设置下拉菜单的 CSS 样式实现半透明效果。实现下拉菜单的半透明效果可以应用 CSS 样式的透明效果滤镜 alpha 实现。alpha 属性是把一个目标元素与背景混合。这种“与背景混合”通俗地说是一个元素的透明度，透明效果滤镜 alpha 的语法如下：

```
{filter:alpha(opacity=opacity,finishopacity=finishopacity,style=style,startx=startx,starty=starty,finishx=finishx,finishy=finishy)}
```

滤镜 alpha 的各个参数说明如表 8.9 所示。

表 8.9 alpha 滤镜的各个参数及说明

属 性	说 明
opacity	代表透明度水准。默认的范围是0~100，其实是百分比的形式，也就是0代表完全透明，100代表完全不透明
finishopacity	可选，如果想要设置渐变的透明效果，可以使用该参数指定结束时的透明度。范围也是0~100
Style	指定透明区域的形状特征，其中0代表统一形状、1代表线形、2代表放射状、3代表长方形
Startx	代表渐变透明效果的开始X坐标
Starty	代表渐变透明效果的开始Y坐标
Finishx	代表渐变透明效果结束X坐标
Finishy	代表渐变透明效果结束Y坐标

在本实例中的 menuskinner 类中，设置了 alpha 滤镜的属性。代码如下：

例程12 代码位置：光盘\TM\08\tmlog\css\style.css

```
.menuskinner {
    BORDER: #666666 1px solid;           //层边框样式
    VISIBILITY: hidden;                  //层可见为隐藏
    FONT: 12px Verdana;                 //层中的字体样式和大小
```

```

POSITION: absolute;           //定位方式
background-image:url("../images/item_out.gif"); //背景图片
background-repeat : repeat-y; //图片是否可重复
Filter: Alpha(Opacity=85);    //设置alpha的opacity属性等于85，即透明度为85%
}

```

在显示页面中显示菜单项的表单引入这个 `menuskin` 类，当鼠标划过特定的文字时，将会发现自动下拉菜单的效果为半透明的样式。页面关键代码如下：

例程13 代码位置：光盘\TM\08\tmlog\file.php

```

<!-- 弹出层设置 -->
<div class=menuskin id=popmenu
  <!-- 鼠标进入该区域时调用的js方法 -->
  onmouseover="clearhidemenu();highlightmenu(event,'on')"
  <!-- 鼠标离开该区域时调用的js方法 -->
  onmouseout="highlightmenu(event,'off');dynamichide(event)"
  <!-- 设置层叠顺序和定位方式 -->
  style="Z-index:100;position:absolute;">
</div>
...
<!-- 半透明的动态下拉菜单 -->
<TABLE style="WIDTH: 580px" VERTICAL-ALIGN: text-top; cellSpacing=0 cellPadding=0 align="center">
  <TR align="center" valign="middle">
    <!-- 显示登录用户 -->
    <TD style="WIDTH: 100px; COLOR: red;">欢迎您:<?php echo
$_SESSION[username]; ?>&nbsp;&nbsp;&nbsp;</TD>
    <!-- 显示首页超链接 -->
    <TD style="WIDTH: 80px; COLOR: red;"><SPAN style="FONT-SIZE: 9pt; COLOR: #cc0033">
</SPAN><a href="index.php">博客首页</a></TD>
    <!-- 显示文章管理下拉菜单 -->
    <TD style="WIDTH: 80px; COLOR: red;"><a onmouseover=showmenu(event,productmenu)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand">文章管理</a></TD>
    <!-- 显示图片管理下拉菜单 -->
    <TD style="WIDTH: 80px; COLOR: red;"><a onmouseover=showmenu(event,Honourmenu)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand">图片管理</a></TD>
    <!-- 显示朋友圈管理下拉菜单 -->
    <TD style="WIDTH: 90px; COLOR: red;"><a onmouseover=showmenu(event,myfriend)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand">朋友圈管理</a></TD>
  <?php
    /* 如果登录用户为管理员 */
    if($_SESSION[fig]==1){
  ?>
    <!-- -----显示管理员管理下拉菜单----- -->
    <TD style="WIDTH: 80px; COLOR: red;"><a onmouseover=showmenu(event,myuser)
onmouseout=delayhidemenu() class='navlink' style="CURSOR:hand">管理员管理</a></TD>
  <?php
    }
  ?>
    <!-- -----显示退出超链接----- -->
    <TD style="WIDTH: 80px; COLOR: red;"><a href="safe.php">退出登录</a></TD>
  </TR>
</TABLE>

```



通过代码可以发现，在每个超链接标签里，都有 3 个参数，分别是：

- ☑ **onMouseOver**：鼠标进入该区域时的效果。本页面调用了 js 方法 showmenu()。
- ☑ **onMouseOut**：鼠标移出该区域时的效果。本页面调用 js 方法 delayhidemenu()。
- ☑ **style**：标签样式，本页面样式为"CURSOR:hand;"，显示一个手的形状。

调用的 js 方法就是实现下拉菜单的 JavaScript 代码：显示和隐藏菜单。由于这段代码是通用型代码，所以单独放到一个 js 文件中，方便其他页调用，js 文件名为 menu.js。调用时在页面中加入如下代码即可。

```
<script src="JS/menu.JS"></script>
```

menu.js 文件中显示、隐藏菜单和变换背景等关键函数的代码如下：

例程14 代码位置：光盘\TM\08\tml\js\menu.js

```
/* 浏览器版本号检测 */
var IE4=document.all&&navigator.userAgent.indexOf("Opera")==-1
var netscape6=document.getElementById&&!document.all
var netscape4=document.layers
/* 显示下拉菜单函数 */
function showmenu(e,vmenu,mod){
/* 判断浏览器版本 */
if (!document.all&&!document.getElementById&&!document.layers)
return
/* 菜单内容 */
which=vmenu
/* 调用clearhidemenu()函数 */
clearhidemenu()
/* 调用IE_clearshadow()函数 */
IE_clearshadow()
/* 根据不同的浏览器，声明弹出式菜单 */
menuobj=IE4? document.all.popmenu : netscape6? document.getElementById("popmenu") : netscape4?
document.popmenu : ""
menuobj.style=(IE4||netscape6)? menuobj.style : menuobj
if (IE4||netscape6)
/* 如果是IE4或netscape6浏览器，就调用innerHTML方法输出下拉菜单 */
menuobj.innerHTML=which
else{
/* 否者使用write方法输出下拉菜单 */
menuobj.document.write('<layer name="other" bgColor="#E6E6E6" width="165"
onmouseover="clearhidemenu()" onmouseout="hidemenu()">'+which+'</layer>')
/* 关闭数据流 */
menuobj.document.close()
}
/* 调用不同的方法属性，来设置不同的浏览器的宽、高、及内部表格的大小、位置 */
menuobj.contentwidth=(IE4||netscape6)? menuobj.offsetWidth : menuobj.document.other.document.width
menuobj.contentheight=(IE4||netscape6)? menuobj.offsetHeight : menuobj.document.other.document.height
eventX=IE4? event.clientX : netscape6? e.clientX : e.x
eventY=IE4? event.clientY : netscape6? e.clientY : e.y
var rightedge=IE4? document.body.clientWidth-eventX : window.innerWidth-eventX
var bottomedge=IE4? document.body.clientHeight-eventY : window.innerHeight-eventY
```

```

        if (rightedge<menuobj.contentwidth)
            menuobj.thestyle.left=IE4? document.body.scrollLeft+eventX-menuobj.contentwidth+menuOffX :
netscape6? window.pageXOffset+eventX-menuobj.contentwidth : eventX-menuobj.contentwidth
        else
            menuobj.thestyle.left=IE4? IE_x(event.srcElement)+menuOffX : netscape6? window.pageXOffset+eventX :
eventX
        if (bottomedge<menuobj.contentheight&&mod!=0)
            menuobj.thestyle.top=IE4?
document.body.scrollTop+eventY-menuobj.contentheight-event.offsetY+menuOffY-23 : netscape6?
window.pageYOffset+eventY-menuobj.contentheight-10 : eventY-menuobj.contentheight
        else
            menuobj.thestyle.top=IE4? IE_y(event.srcElement)+menuOffY : netscape6?
window.pageYOffset+eventY+10 : eventY
    /* 设置标签样式为可见 */
    menuobj.thestyle.visibility="visible"
    /* 调用IE_dropshadow()方法 */
    IE_dropshadow(menuobj,"#999999",3)
    return false
}
...
/* 隐藏下拉菜单函数 */
function hidemenu(){
    if (window.menuobj)
        menuobj.thestyle.visibility=(IE4||netscape6)? "hidden" : "hide"
    IE_clearshadow()
}
function dynamichide(e){
    if (IE4&&!menuobj.contains(e.toElement))
        hidemenu()
    else if (netscape6&&e.currentTarget!= e.relatedTarget&& !contains_netscape6(e.currentTarget, e.relatedTarget))
        hidemenu()
}
/* 延迟隐藏下拉菜单函数 */
function delayhidemenu(){
    if (IE4||netscape6||netscape4)
        delayhide=setTimeout("hidemenu()",500)
}
/* 停止隐藏菜单函数 */
function clearhidemenu(){
    if (window.delayhide)
        clearTimeout(delayhide)
}
...
/* 设置菜单背景 */
/* 鼠标移入该区域时显示的背景 */
function overbg(tdbg){
    tdbg.style.background='url( images/item_over.gif)'
    tdbg.style.border=' #9CA6C6 1px solid'
}
/* 鼠标移出该区域时显示的背景 */
function outbg(tdbg){

```



```

tdbg.style.background='url( images/item_out.gif)'
tdbg.style.border="
}
/* 下拉菜单内容 */
/* 文章管理下拉菜单 */
var productmenu='<table width=90><tr><td id=fileadd onMouseOver=overbg(fileadd) onMouseOut=outbg(fileadd)><a
href=file.php>添加博客文章</a></td></tr>\
<tr><td id=query onMouseOver=overbg(query) onMouseOut=outbg(query)><a href=query.php>查询博客文章
</a></td></tr>\
<tr><td id=myfiles onMouseOver=overbg(myfiles) onMouseOut=outbg(myfiles)><a href=myfiles.php>我的文章
</a></td></tr></table>'
/* 图片管理下拉菜单 */
var Honourmenu='<table width=90><tr><td id=picadd onMouseOver=overbg(picadd) onMouseOut=outbg(picadd)><a
href=add_pic.php>添加图片</a></td></tr>\
<tr><td id=browse onMouseOver=overbg(browse) onMouseOut=outbg(browse)><a href=browse_pic.php>浏览图片
</a></td></tr>\
<tr><td id=querpypic onMouseOver=overbg(querpypic) onMouseOut=outbg(querpypic)><a href=query_pic.php>查询图片
</a></td></tr></table>'
/* 朋友圈管理下拉菜单 */
var myfriend='<table width=90><tr><td id=friendadd onMouseOver=overbg(friendadd)
onMouseOut=outbg(friendadd)><a href=friend.php>添加到朋友圈</a></td></tr>\
<tr><td id=browse_fri onMouseOver=overbg(browse_fri) onMouseOut=outbg(browse_fri)><a href=browse_fri.php>浏览
我的朋友</a></td></tr>\
<tr><td id=cxfriend onMouseOver=overbg(cxfriend) onMouseOut=outbg(cxfriend)><a href=query_friend.php>查询朋友
信息</a></td></tr></table>'
/* 管理员管理下拉菜单 */
var myuser='<table width=90><tr><td id=queryuser onMouseOver=overbg(queryuser)
onMouseOut=outbg(queryuser)><a href=queryuser.php>查询用户信息</a></td></tr>\
<tr><td id=browseuser onMouseOver=overbg(browseuser) onMouseOut=outbg(browseuser)><a href=browseuser.php>浏
览用户信息</a></td></tr>\
<tr><td id=managepub onMouseOver=overbg(managepub) onMouseOut=outbg(managepub)><a href=managepub.php>
公告管理</a></td></tr>\
</table>'

```

8.10 登录验证码技术专题

当今网络安全越来越成为一个重要的问题，开发一个带验证码的用户登录模块可以提高网站的安全性。其中，验证码是随机生成的，可以是字母、数字或汉字，也可以是图片。本节技术专题主要就是学习一下验证码的几种实现方法及技巧。

8.10.1 简单的数字验证

先来看一个简单的数字验证码：生成一组 4 位的随机数字，每刷新一次，显示的结果都不相同。如果验证码输入错误，将弹出提示窗口。数字验证码运行结果如图 8.31 所示。



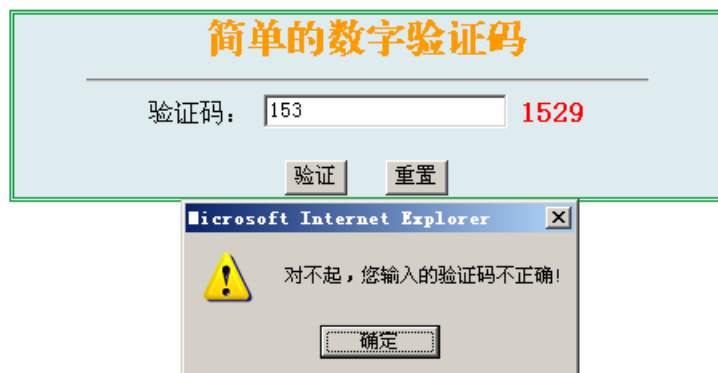


图 8.31 简单的数字验证运行结果

简单的数字验证码在开发过程中主要应用两个函数：

(1) mt_rand()函数

mt_rand()函数主要用于从指定参数中随机去一个数值。语法如下：

```
int mt_rand([int min], [int max]);
```

例如：从 52~79 之间取一个随机数值。代码如下：

```
mt_rand(52,79)
```

(2) intval()函数

intval()函数主要用于将变量转成整数类型。语法如下：

```
int intval(mixed var, int [base]);
```

intval()函数可将变量转成整数类型。可省略的参数 base 是转换的基数，默认值为 10。转换的变量 var 可以为数组或类之外的任何类型变量。

8.10.2 数字图形验证码

数字图形验证码相对于数字验证码而言，要稍复杂一些。数字图形验证码主要应用 mt_rand 函数来初始化一组 4 位的随机数，然后利用 for 循环语句将随机生成的 4 位随机验证码利用数字图形进行输出。图形验证码的运行结果如图 8.32 所示。

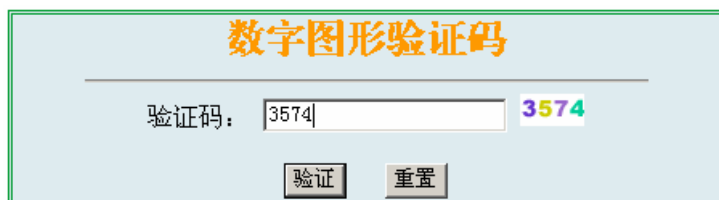


图 8.32 数字图形验证码

实现数字图形验证码的关键代码如下：




```

<!-- js脚本，判断文本框中输入的验证码是否匹配 -->
<script language="javascript">
    function check(form){
        if(form.txt_yan.value==""){
            alert("请输入验证码");form.txt_yan.focus();return false;
        }
        if(form.txt_yan.value!=form.txt_hyan.value){
            alert("对不起，您输入的验证码不正确!");form.txt_yan.focus();return false;
        }
    }
</script>
...
<!-- 验证表单 -->
<form name="form" method="post" action="" onSubmit="return check(form);">
<!-- 验证码文本框 -->
<input type="text" name="txt_yan">
<!-- 生成随机数字图形验证码 -->
<?php
<!-- 生成随机4位数 -->
    $num=intval(mt_rand(1000,9999));
<!-- 使用for循环，输出随机4位数的图片格式 -->
    for($i=0;$i<4;$i++){
        echo "<img src=images/checkcode/".substr(strval($num),$i,1).".gif>";
    }?>
<!-- 将随机数存入隐藏域 -->
<input type="hidden" name="txt_hyan" id="txt_hyan" value="<?php echo $num;?>">
<br> <br>
<input type="submit" name="Submit" value="验证"> &nbsp; <input type="reset" name="Submit2" value="重置">
<!-- 处理代码段 -->
<?php
/* 判断验证码录入框是否为空 */
    if($_POST[txt_yan]!=""){
/* 当输入的验证码和隐藏域中的验证码相等时，输出欢迎信息 */
        if($_POST[txt_yan] == $_POST[txt_hyan]){
            echo "您输入的验证码通过，感谢您的加盟...";
        }
    }
?>

```

8.10.3 汉字图形验证码

汉字图形验证码是目前网络上比较流行的验证码方式，这种验证码的实现有一定的难度。首先将图片所对应的汉字依次存储到数组中。然后利用 Rand 函数生成一个小于 10 的数，用于得到随机字符的位置，进而得出随机字符，并设置要显示的随机图片。由于产生的随机字符串是 4 位的，所以需要 使用 for 循环语句，根据数组得出相应图片的路径，再将图片转换成数组中的文字，最后输出到浏览器中。

以汉字图形化显示验证码的完整代码如下：




```
<?php
$str=array("大","更","创","天","科","客","博","技","立","新");
$word=strlen($str);
for($i=0;$i<4;$i++){
    $num=rand(0,$word*2-1);
    $img=$img."<img src=' images/checkcode/'".$num.".gif' width='16' height='16'>"; //显示随机图片
    $pic=$pic.$str[$num]; //将图片转换成数组中的文字
}
?>
```

8.11 本章总结

本章的博客管理系统首先介绍了博客的基本概念、发展前景、影响范围及博客网的功能分类，使读者对当今主流博客有了一个大致的认识。其次，实现了一个博客系统、包含所有基本功能的项目开发，使读者对如何开发一个博客网站有了一个初步的了解。最后，希望读者通过自己的努力，来逐步完善和加强这个博客网的实用功能，最终达到一个令自己满意的作品。

