PHP实现简单的聊天室应用

# 开发环境

1、环境搭建：Windows 7+Apache 2.4.18+MySQL 5.7.11+PHP 7.1.0。

2、文本编辑器：Sublime 3。

# 二、主要技术

本案例使用PHP 7中的PDO数据对象对MySQL数据库进行查询、插入、修改和删除操作，同时使用Ajax技术实现无刷新页面发送和接收消息，从而实现简单网页聊天室应用的制作。

# 三、效果图展示

实现效果图，如图1、图2所示。

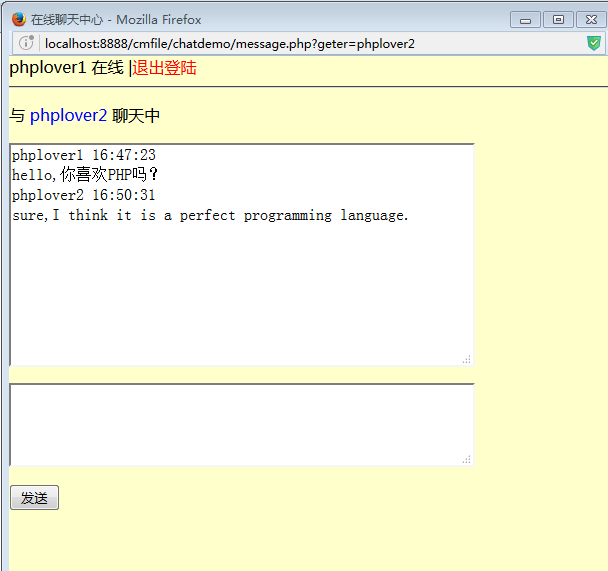
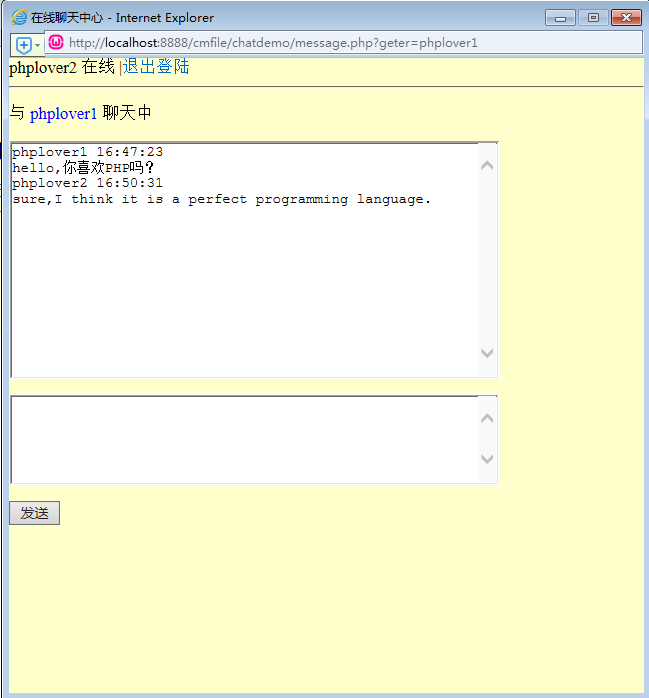


图1 效果图（用户“phplover1”与用户“phplover2”聊天页面）

  
图2 效果图（用户“phplover2”与用户“phplover1”聊天页面）

# 四、具体步骤

## 4.1、数据库设计

### （1）创建数据库

创建一个名为“phpDemo”的数据库。

### （2）创建数据表

①创建“041\_1”表

“041\_1”表用来存储用户信息，其表结构如图3所示：



图3 数据库表（“041\_1”表）结构

②创建“041\_2”表

“041\_2”表用来存储好友信息，其表结构如图4所示：

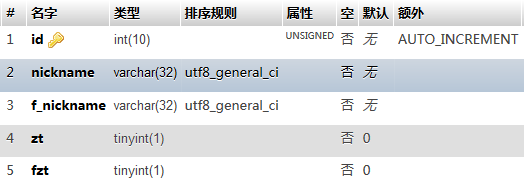


图4 数据库表（“041\_2”表）结构

③创建“041\_3”表

“041\_3”表用来存储聊天信息，其表结构如图5所示：



图5 数据库表（“041\_3”表）结构

### （3）向各个表中添加数据

## 4.2、连接数据库

### （1）创建PHP文件

创建一个名为“conn.php”的PHP文件。

### （2）连接数据库

<?php

/\*\*

\*使用PDO方式连接数据库

\*/

$dbms='mysql'; //数据库类型

$host='localhost:3307'; //数据库主机名

$dbName='phpDemo'; //使用的数据库

$user='root'; //数据库连接用户名

$pass=''; //对应的密码

$dsn="$dbms:host=$host;dbname=$dbName";

try {

$dbh = new PDO($dsn, $user, $pass); //初始化一个PDO对象

$dbh->query('SET NAMES utf8'); // 设置字符集

} catch (PDOException $e) {

die ("Error!: " . $e->getMessage() . "<br/>");

}

?>

## 4.3、用户登录

### （1）界面设计

①创建PHP文件

创建一个名为“login.php”的PHP文件。

②表单设计

<!-- 登录界面 -->

<!DOCTYPE html>

<html>

<head>

<title>在线聊天程序用户登录中心</title>

<meta charset="utf-8">

<link rel="stylesheet" type="text/css"

href="./css/login\_style.css">

</head>

<body>

<div id="denglu">

<div id="top">

<b>在线聊天程序用户登录中心</b>

</div>

<div id="bottom">

<form action="login\_check.php" method="post">

<table>

<tr>

<td>用户名：</td>

<td><input type="text" name="name" class="txt"></td>

</tr>

<tr>

<td>密 &nbsp;&nbsp; 码：</td>

<td><input type="password" name="password"

class="txt"></td>

</tr>

<tr>

<td>验证码：</td>

<td>

<input type="text" name="yzm" size="10"

class="yzm">

<a href="login.php"><image

src="./yanzhengma.php"></a>

</td>

</tr>

<tr>

<td colspan="2"><center>

<input type="submit" name="sub" value="登录

">&nbsp;&nbsp;

<input type="reset" name="ret" value="重置

">&nbsp;&nbsp;

<a href="zhuce.php">立即注册</a>

</center></td>

</tr>

</table>

</form>

</div>

</div>

</body>

</html>

③样式设计（CSS样式）

body{

cellpadding:0;

cellspacing:0;

margin:0;

padding:0;

background-color: white;

}

#denglu{

width:460px;

margin:50px auto;

border:1px solid blue;

border-radius:10px;

background-color: #FFFFCC;

}

#top{

text-align:center;

padding-top:20px;

line-height:2em;

color:blue;

font-size:24px;

font-family:'宋体';

}

#bottom table{

width:280px;

height:200px;

margin: 20px auto;

}

#bottom table .txt{

width:160px;

border:1px solid gray;

}

#bottom table .yzm{

border:1px solid gray;

}

#bottom table caption{

font-size:18px;

font-weight:300;

font-family:'宋体';

}

#bottom table a{

color:gray;

font-size:12px;

text-decoration: none;

}

#bottom table a:hover{

color:red;

}

#zhuce{

width:460px;

margin:50px auto;

border:1px solid blue;

border-radius:10px;

background-color: #CCFFCC;

padding-bottom:20px;

}

### （2）实现登录功能

①创建PHP文件

创建一个名为“login\_check.php”的PHP文件用来验证用户登录。

②验证用户登录

if(isset($\_POST['sub'])){

// 获得用户输入信息

$name=$\_POST['name']; // 用户名

$password=md5($\_POST['password']); // 用户输入的密码并对其

进行md5加密

$yzm=$\_POST['yzm']; // 用户输入的验证码

@session\_start(); // 开启session

$checkstr=$\_SESSION['string']; //获取验证码字符串

// 判断输入是否为空

if(empty($name)||empty($password)||empty($yzm)){

echo "<script>alert('输入不能为空！');

history.back();</script>";

}else if(!strcasecmp($yzm, $checkstr)==0){ // 判断验证码是 否输入正确(不区分大小写)

echo "<script>alert('验证码输入错误！');

history.back();</script>";

}else{

// 判断用户信息是否正确

require("./conn.php");

$sql="SELECT \* FROM `041\_1` WHERE `nickname`='$name'

AND `password`='$password'";

$result=$dbh->query($sql); // 执行查询

if($result->rowCount()){ // 判断用户信息是否存在

// 保存用户信息

$\_SESSION['nickname']=$name; // 用户名

$\_SESSION['password']=$password; //密码

// 输出登录成功提示信息

echo "<script>alert('登录成功！

');location.href='index.php'</script>";

}else{

// 输出登录失败提示信息

echo "<script>alert('用户名或密码错误！');

history.back();</script>";

}

}

}

## 4.4、用户注册

### （1）界面设计

①创建PHP文件

创建一个名为“zhuce.php”的PHP文件。

②表单设计

<!-- 注册界面 -->

<!DOCTYPE html>

<html>

<head>

<title>在线聊天程序用户注册中心</title>

<meta charset="utf-8">

<link rel="stylesheet" type="text/css"

href="./css/login\_style.css">

</head>

<body>

<div id="zhuce">

<div id="top">

<b>在线聊天程序用户注册中心</b>

</div>

<div id="bottom">

<form action="zhuce\_action.php" method="post">

<table>

<tr>

<td>用户名：</td>

<td><input type="text" name="name" class="txt"></td>

</tr>

<tr>

<td>住 &nbsp;&nbsp;址：</td>

<td><input type="text" name="address"

class="txt"></td>

</tr>

<tr>

<td>性 &nbsp;&nbsp;别：</td>

<td>

<input type="radio" name="sex" value="男" checked>

男

<input type="radio" name="sex" value="女">女

</td>

</tr>

<tr>

<td>出生日期：</td>

<td>

<input type="text" name="year" size="4"

value="1995">年

<select name="month">

<?php

for($i=1;$i<=12;$i++){

echo "<option>$i</option>";

}

?>

</select>月

<select name="day">

<?php

for($i=1;$i<=31;$i++){

echo "<option>$i</option>";

}

?>

</select>日

</td>

</tr>

<tr>

<td>密 &nbsp;&nbsp; 码：</td>

<td><input type="password" name="password"

class="txt"></td>

</tr>

<tr>

<td colspan="2"><center>

<input type="submit" name="sub" value="注册

">&nbsp;&nbsp;

<input type="reset" name="ret" value="重置">

</center></td>

</tr>

</table>

</form>

</div>

</div>

</body>

</html>

### （2）实现注册功能

①创建PHP文件

创建一个名为“zhuce\_action.php”的PHP文件用来实现用户注册功能。

②实现用户注册功能

header("Content-type:text/html;charset=utf-8");

if(isset($\_POST['sub'])){

// 获取用户输入信息

$name=$\_POST['name']; // 姓名

$address=$\_POST['address']; // 住址

$sex=$\_POST['sex']; // 性别

$year=$\_POST['year']; //年

$month=$\_POST['month']; //月

$day=$\_POST['day']; //日

$birthday=$year."-".$month."-".$day; // 出生年月

$password=$\_POST['password']; // 密码

$age=intval(date("Y"))-intval($year); // 年龄

$password\_md5=md5($password); // 密码md5加密

// 判断用户名或密码是否为空

if(empty($name)||empty($password)){

echo "<script>alert('请输入用户名或密码！');

history.back();</script>";

exit();

}

// 判断该用户是否已注册

require("./conn.php");

$sql\_select="select \* from `041\_1` where

nickname='".$name."';";

$result\_select=$dbh->query($sql\_select);

if($result\_select->rowCount()){

echo "<script type='text/javascript'> alert('该用户名已

经被注册'); history.back();</script>";

exit();

}

// 注册功能

$sql\_insert="insert into `041\_1` (id,nickname,password,address,sex,age,birthday,reg\_time)

values(NULL,'$name','$password\_md5','$address','$sex','$age','$birthday',now())";

$result\_insert=$dbh->query($sql\_insert);

if($result\_insert->rowCount()){

echo "<script>alert('恭喜！注册成功!请前往登录中心！

');location.href='login.php';</script>";

}else{

echo "<script type='text/javascript'> alert('对不起！注

册失败！');history.back(); </script>";

}

}

## 4.5、聊天室主页面

### （1）创建PHP文件

创建一个名为“index.php”的PHP文件。

### （2）显示当前用户和好友信息

<div id="main">

<h3><center>在线聊天中心</center></h3>

<?php

// 显示当前用户和好友请求信息

session\_start();

require("./conn.php");

// 判断用户是否已登录

if(empty($\_SESSION['password'])){

header("Location:login.php");

exit();

}else{

$nickname = $\_SESSION['nickname'];

echo "当前用户：<a href='index.php'>".$nickname."</a>";

//判断是否有好友请求

$sql = "select id from `041\_2` where

f\_nickname='{$nickname}' and fzt='0';";

$res = $dbh->query($sql);

$fnum = $res->rowCount();

if($fnum>0){

echo "<span ><a href='qingqiu.php'

style='color:#c00'>&nbsp;您有(".$fnum.")

条好友请求</span></a> 在线<a href='exit.php'>

退出登陆</a>";

}else{

echo " 在线 <a href='exit.php'>退出登陆</a>";

}

}

?>

<hr />

<p><span style="font-weight:bold">好友列表</span>&nbsp;|&nbsp;<a href="addfriend.php">添加好友</a></p>

<ul id="flist">

<?php

$sql = "select f\_nickname from `041\_2` where

nickname='{$nickname}' and fzt='1';";

//echo $sql;exit();

$res =$dbh->query($sql);

if($res->rowCount()<1){

echo "您还没有好友！<a href='addfriend.php'>添加好友

</a>";

exit();

}

$result=$res->fetchAll(PDO::FETCH\_ASSOC);

echo "<table>";

foreach($result as $row){

echo "<tr>";

$f\_nickname = $row['f\_nickname'];

//判断是否有新消息

$sql1 = "select id from `041\_3` where

sender='{$f\_nickname}' and geter='{$nickname}'

and mloop=0;";

$res1 = $dbh->query($sql1);

//echo $sql1;

echo "<td><li

title='".$f\_nickname."'>".$f\_nickname;

if($res1->rowCount()>0){

echo "<span style='color:red'>(有新消

息)</span></li></td>";

}else{

echo "</li></td>";

}

echo "<td>&nbsp;&nbsp;<a

href='delfriend.php?f\_nickname=".$f\_nickname."'

onclick='return del\_confirm()'>删除</a></td>";

echo "</tr>";

}

echo "</table>";

?>

</ul>

</div>

### （3）当点击好友的昵称时跳转至聊天页面

<script type="text/javascript" src="js/jquery-1.7.2.min.js"></script>

<script type="text/javascript">

// 当点击好友的昵称时跳转至聊天页面

$(function(){

$("#flist

li").hover(function(){ $(this).css("color","blue").css

("cursor","pointer");$(this).siblings().css

("color","#000")},function(){$(this).css

("color","#000")}).click(function(){window.open

("message.php?geter="+$(this).attr("title"),

"webchat","width=600,height=600");});

});

</script>

## 4.6、添加好友

### （1）界面设计

①创建PHP文件

创建一个名为“addfriend.php”的PHP文件。

②表单设计

<!-- 添加好友操作界面 -->

<!DOCTYPE html>

<html>

<head>

<title>在线聊天中心</title>

<meta charset="utf-8">

<link rel="stylesheet" type="text/css"

href="./css/style.css">

</head>

<body>

<div id="main">

<P><b>添加好友</b>&nbsp;|&nbsp;<a href="index.php">返回主页 </a></P>

<hr />

<!-- 提交给"addfriend1.php"页面处理 -->

<form action="addfriend1.php" method="get">

好友昵称：<input type="text" name="f\_nickname" />

<input type="submit" value=" 添加 " name="sub" />

</form>

<hr />

<p>最新注册会员列表：</p>

<?php

require("./conn.php"); // 引入数据库连接文件

// 查询user表中所有注册人员的昵称，查询结果显示前10条并按

照注册时间降序排序

$sql = "select nickname from user order by reg\_time

desc limit 0,10;";

$res = $dbh->query($sql); // 执行sql查询

$result=$res->fetchAll(PDO::FETCH\_ASSOC);

// 显示所有注册会员

echo "<div id='allmember'>";

foreach($result as $row){

echo "<li>".$row['nickname']."&nbsp;|&nbsp;<a

href='addfriend\_action.php?f\_nickname=

".$row['nickname']."'>添加好友</a></li>";

}

echo "</div>";

?>

</div>

</body>

</html>

### （2）实现添加好友功能

①创建PHP文件

创建一个名为“addfriend\_action.php”的PHP文件。

②实现添加好友功能

<?php

/\*\*

\*添加好友

\*/

session\_start();

// 如果用户未登录（避免未登录用户直接访问此页）

// target='\_blank'：在新窗口打开标签页。

if(empty($\_SESSION['password'])){

echo "<script>alert('您还没有登录，请登录！');

location.href='login.php';</script>";

exit();

}

header("Content-type:text/html; charset=utf-8"); // 设置文件类型和编码

require("./conn.php"); // 包含进连接数据库文件

$nickname = $\_SESSION['nickname']; // 用户登录昵称

$f\_nickname = $\_GET['f\_nickname']; // 要添加的好友的昵称

//判断要添加的好友是否存在

$sql = "select id from `041\_1` where nickname='{$f\_nickname}';";

$res = $dbh->query($sql);

// 如果不存在，则终止程序不再执行

if($res->rowCount()<1){

echo "<script type='text/javascript'> alert('不存在该用

户'); location.href='addfriend.php'; </script>";

exit();

}

//从friend表中查询判断是否已经加过该好友

$sql2 = "select nickname from `041\_2` where f\_nickname='{$f\_nickname}' and nickname='{$nickname}';";

$res2 = $dbh->query($sql2);

if($res2->rowCount()>0){

echo "<script type='text/javascript'> alert('您已经添加

了该好友'); location.href='addfriend.php'; </script>";

exit();

}

// 添加好友

$sql3= "insert `041\_2` (nickname,f\_nickname) values ('{$nickname}','{$f\_nickname}');";

$res3 = $dbh->query($sql3);

if($res3->rowCount()){

echo "<script type='text/javascript'> alert('好友请求发

送成功，请等待对方确认'); location.href='addfriend.php';

</script>";

}

?>

## 4.7、好友请求处理

### （1）显示好友请求信息

①创建PHP文件

创建一个名为“qingqiu.php”的PHP文件。

②显示好友请求信息

<!-- 显示好友请求信息 -->

<?php session\_start(); ?>

<!DOCTYPE html>

<html>

<head>

<title>在线聊天中心</title>

<meta charset="utf-8">

<link rel="stylesheet" type="text/css"

href="./css/style.css">

</head>

<body>

<div id="qingqiu">

<?php

require("conn.php");

if(empty($\_SESSION['password'])){

header("Location:login.php");

exit();

}else{

$nickname = $\_SESSION['nickname'];

echo "<div id='bar'>";

echo "<a href='index.php'>".$nickname." 的主页

</a>";

//判断是否有好友请求

$sql = "select id from `041\_2` where

f\_nickname='{$nickname}' and fzt='0';";

$res = $dbh->query($sql);

$fnum = $res->rowCount();

if($fnum>0){

echo "<span ><a href='qingqiu.php'

style='color:#c00'>&nbsp;您有(".$fnum.")

条好友请求</span></a> 在线

<a href='exit.php'>退出登陆</a>";

}else{

echo " 在线 <a href='exit.php'>退出登陆

</a>";

}

echo "</div>";

}

?>

<div id="message">

<hr />

<p><b>好友请求</b></p>

<?php

$sql = "select id,nickname,f\_nickname from `041\_2`

where f\_nickname='{$nickname}' and fzt='0';";

$res = $dbh->query($sql);

if($res->rowCount()<1){

echo "没有好友请求";

exit();

}

$result=$res->fetchAll(PDO::FETCH\_ASSOC);

foreach($result as $row){

echo "<p style='float:left;margin-left:30px;'><span

style='color:#00a;font-weight:bold;'>";

echo $row['nickname']."</span> 请求加为好友&nbsp;<a

href='agreeqingqiu.php?f\_nickname=";

echo $row['nickname']."&id=".$row['id']."'>同意并添

加</a>&nbsp;<a href='refuseqingqiu.php?

id=".$row['id']."'>拒绝</a></p>";

}

?>

</div>

</body>

</html>

### （2）同意添加对方为好友

①创建PHP文件

创建一个名为“agreeqingqiu.php”的PHP文件。

②实现同意添加对方为好友的功能

<?php

/\*\*

\*同意添加对方为好友

\*/

session\_start();

if(empty($\_SESSION['password'])){

echo "<a href='login.php'>登陆</a> <a href='regist.php'

target='\_blank'>注册</a>";

exit();

}

header("Content-type:text/html; charset=utf-8");

require("conn.php");

$id = $\_GET['id'];

$f\_nickname = $\_GET['f\_nickname'];

//同意添加

$sql = "update `041\_2` set fzt=1 where id='{$id}';";

$res = $dbh->query($sql);

//并将对方加为好友

$nickname = $\_SESSION['nickname'];

//如果不是自己加自己则进行这步

if($f\_nickname!=$nickname){

$sql = "insert into `041\_2` (nickname,f\_nickname,fzt)

value ('{$nickname}','{$f\_nickname}','1');";

$res = $dbh->query($sql);

}

if($res->rowCount()){

echo "<script type='text/javascript'> alert('操作成功

'); location.href='qingqiu.php'; </script>";

exit();

}

?>

### （3）拒绝添加对方为好友

①创建PHP文件

创建一个名为“refuseqingqiu.php”的PHP文件。

②实现拒绝添加对方为好友的功能

<?php

/\*\*

\*拒绝好友请求

\*/

session\_start();

if(empty($\_SESSION['password'])){

echo "<a href='login.php'>登陆</a> <a href='regist.php'

target='\_blank'>注册</a>";

exit();

}

header("Content-type:text/html; charset=utf-8");

require("conn.php");

$id = $\_GET['id'];

$sql = "delete from `041\_2` where id='{$id}';";

$res = $dbh->query($sql);

if($res){

echo "<script type='text/javascript'> alert('已拒绝');

location.href='qingqiu.php'; </script>";

exit();

}

?>

## 4.8、删除好友

①创建PHP文件

创建一个名为“delfriend.php”的PHP文件。

②实现删除好友功能

<?php

/\*\*

\*删除好友

\*/

session\_start();

// 如果用户未登录（避免未登录用户直接访问此页）

if(empty($\_SESSION['password'])){

header("Location:login.php");

exit();

}

header("Content-type:text/html; charset=utf-8");

require("./conn.php");

$f\_nickname = $\_GET['f\_nickname']; // 要删除的好友昵称

$sql = "delete from `041\_2` where nickname='".$\_SESSION['nickname']."' and f\_nickname='{$f\_nickname}'";

$stmt=$dbh->query($sql); // 执行删除

// 判断是否删除成功

if($stmt->rowCount()){

echo "<script type='text/javascript'> alert('删除成功

'); location.href='index.php'; </script>";

}else{

echo "<script type='text/javascript'> alert('删除失败

'); location.href='index.php'; </script>";

}

?>

## 4.9、聊天功能

### （1）聊天主页面

①创建PHP文件

创建一个名为“message.php”的PHP文件。

②界面设计

<body style="background-color:#FFFFCC">

<?php

session\_start();

require("conn.php");

$geter= $\_GET['geter']; // 好友昵称

$nickname = $\_SESSION['nickname']; // 当前用户昵称

if(empty($\_SESSION['password'])){

echo "<a href='login.php'>登陆</a> <a

href='regist.php' target='\_blank'>注册</a>";

exit();

}else{

echo $nickname. " 在线 |<a href='exit.php'>退出登陆

</a>";

}

?>

<div id="message">

<hr />

<p>与<font color='blue'> <?php echo $geter; ?> </font>聊天中</p>

<textarea readonly="readonly" id="messageBox"></textarea>

</div>

<div id="message2">

<textarea name="content" id="sendBox"></textarea>

<p><input type="button" value="发送" id="sendmess"

/></p>

</div>

</body>

③创建Ajax引擎对象

function createAjaxObject(){

if(window.ActiveXObject){

var newRequest = new

ActiveXObject("Microsoft.XMLHTTP");

}else{

var newRequest = new XMLHttpRequest();

}

return newRequest;

}

④应用Ajax技术实现无刷新消息发送

// 当点击发送按钮时触发sendMessage函数

$(function(){

$("#sendmess").click(sendMessage);

});

// 发送消息函数（应用Ajax技术无刷新实现消息发送）

function sendMessage(){

var http\_request = createAjaxObject();

if(http\_request){

var url = "sendmes.php"; // 文件在服务器上的位置

var sender = "<?php echo $nickname; ?>"; // 发送者昵称

var geter = "<?php echo $geter; ?>"; // 接收者昵称

var content = $("#sendBox").val(); // 要发送的消息内容

var data = "content="+content+"&sender="+sender+

"&geter="+geter;

// alert(data);

http\_request.open("post",url,true); // 规定请求的类型为

post、url地址、异步处理

http\_request.setRequestHeader("content-

type","application/x-www-form-urlencoded");

http\_request.onreadystatechange = function(){

if(http\_request.readyState==4){

//等于200表示成功

if(http\_request.status==200){

var res = http\_request.responseText;

if(res!=""){

//res==""说明发送成功，然后就将发送信息写入

messageBox

//var nowtime = new Date().toLocaleString();

var content1 = "<?php echo $nickname.'

'; ?>"+res+"\r\n"; // 发送者和发送时间

var content2 = content+"\r\n" ;

var contents = $("#messageBox").val()+

content1+content2;

// alert(content);

$("#messageBox").val(contents);

$("#sendBox").val(""); //将发送框清除

}

}

}

}

http\_request.send(data); // 将请求发送到服务器

}

}

⑤应用Ajax技术实现无刷新消息接收

setInterval(getMessage,1000); //每1秒发送一次请求

function getMessage(){

var http\_request = createAjaxObject();

if(http\_request){

var url = "getmes.php";

var sender = "<?php echo $nickname; ?>"; // 发送方昵称

var geter = "<?php echo $geter; ?>"; // 接收方昵称

var data = "sender="+sender+"&geter="+geter;

// alert(data);

http\_request.open("post",url,true);

http\_request.setRequestHeader("content-

type","application/x-www-form-urlencoded");

http\_request.onreadystatechange = function(){

if(http\_request.readyState==4){

//等于200表示成功

if(http\_request.status==200){

if(http\_request.responseText=="nomessage")

{return;}

var res = eval("("+

http\_request.responseText+")");

for(var i=0;i<res.length;i++){

var content1 = "<?php echo $geter; ?> "+

res[i].stime+"\r\n";

var content2 = res[i].content+"\r\n" ;

var contents = $("#messageBox").val()+

content1+content2;

$("#messageBox").val(contents);

}

}

}

}

http\_request.send(data);

}

}

### （2）发送消息

①创建PHP文件

创建一个名为“sendmes.php”的PHP文件。

②实现消息发送功能

<?php

/\*\*

\*发送消息

\*/

header("Content-type:text/html;charset=utf-8");

if(empty($\_POST['sender'])){exit();}

$sender = $\_POST['sender'];

$geter = $\_POST['geter'];

$content = $\_POST['content'];

require("conn.php");

$sql = "INSERT INTO `041\_3`

(sender,geter,content,stime) VALUES

('{$sender}','{$geter}','{$content}',now())";

$res = $dbh->query($sql);

if($res->rowCount()==0){

echo ""; //发送失败

}else{

date\_default\_timezone\_set("PRC");

echo date("H:i:s");

}

?>

### （3）接收消息

①创建PHP文件

创建一个名为“getmes.php”的PHP文件。

②实现消息接收功能

<?php

/\*\*

\*接收消息

\*/

header("Content-type:text/html;charset=utf-8");

if(empty($\_POST['sender'])){exit();}

$sender = $\_POST['sender'];

$geter = $\_POST['geter'];

require("conn.php");

$sql = "select content,stime from `041\_3` where sender='{$geter}' and geter='{$sender}' and mloop=0 order by stime asc";

$res =$dbh->query($sql);

$row=$res->fetchAll(PDO::FETCH\_ASSOC);

$cont=$row[0]['content'];

$stime=$row[0]['stime'];

//消息条数

$mNums = $res->rowCount();

if($mNums<1){

echo "nomessage";

exit();

}else if($mNums==1){

echo "[{'content':'".$cont."','stime':

'".substr($stime,11)."'}]";

}else{

$result="[{'content':'".$cont."',

'stime':'".substr($stime,11)."'}";

$result1=$res->fetchAll(PDO::FETCH\_ASSOC);

foreach($result1 as $row){

$cont=$row['content'];

$stime=$row['stime'];

$result.=",{'content':'".$cont."',

'stime':'".substr($stime,11)."'}";

}

$result.=']';

echo $result;

}

//收到消息后将它的状态设为1

if($mNums>0){

$sql = "update `041\_3` set mloop=1 where

sender='{$geter}' and geter='{$sender}'

and mloop=0";

$dbh->query($sql);

}

?>

## 4.10、退出登录

### （1）创建PHP文件

创建一个名为“exit.php”的PHP文件。

### （2）实现退出登录功能

<?php

/\*\*

\*退出系统

\*/

session\_start();

unset($\_SESSION['nickname']);

unset($\_SESSION['password']);

header("Location:index.php");

?>

# 五、网页测试

①开启Apache和MySQL服务，打开浏览器，在浏览器地址栏中输入文件地址，例如:“localhost:8080/cmfile/chatdemo/login.php”，可以看到程序运行结果如图6所示。

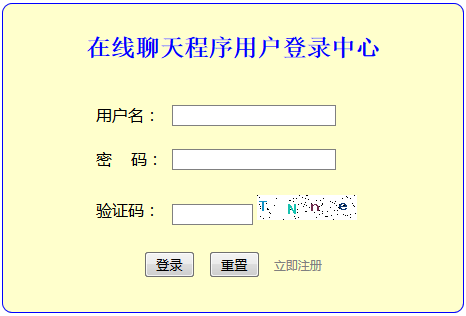


图6 案例结果图（用户登录界面）

②点击【注册】，可以看到页面跳转到注册页面,如图7所示。



图7 案例结果图（用户注册界面）

③输入用户名“phplover1”、住址“北京”、性别“女”、出生日期“1995”年“10”月“5”日和密码“123”，进行注册，如图8所示。



图8 案例结果图（进行用户注册）

④点击【注册】，可以看到如图9所示的提示信息。

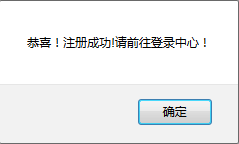


图9 案例结果图（用户注册成功）

⑤然后点击【确定】，可以看到页面跳转到登录页面，我们输入刚刚注册好的用户名和密码，点击【登录】，可以看到如图10所示的登录成功的提示信息。

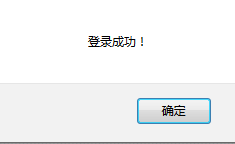


图10案例结果图（用户登录成功）

⑥点击【确定】，可以看到页面跳转到聊天主页面，如图11所示。

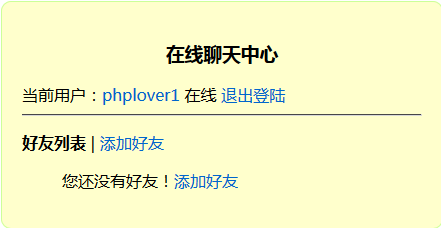


图11 案例结果图（聊天主页面）

⑦打开另一个浏览器（因为同一个浏览器会有缓存，不能实现两个用户同时登录），输入文件地址后再注册一个账号。例如注册一个用户名为“Phplover2”、密码为“123”的用户。然后注册成功后登录。

⑧此时在用户“phplover1”的聊天主页面上点击“添加好友”的超连接，可以看到页面跳转到添加好友页面，如图12所示。



图12 案例结果图（添加好友页）

⑨此时可以看到新注册会员里面有用户“phplover2”，点击“phplover2”后面的“添加好友”的超连接，可以看到如图13所示的添加好友请求发送成功提示信息。

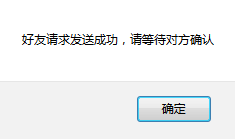


图13 案例结果图（添加好友请求发送成功提示信息）

⑩此时打开用户“phplover2”的聊天主页面，可以看到如图14所示的好友请求提示信息。

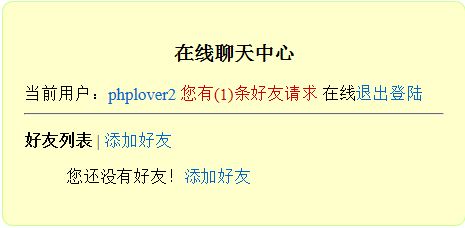


图14 案例结果图（好友请求提示信息）

⑪点击该页面中的“您有（1）条好友请求”的超链接，可以看到页面跳转到如图15所示的页面。

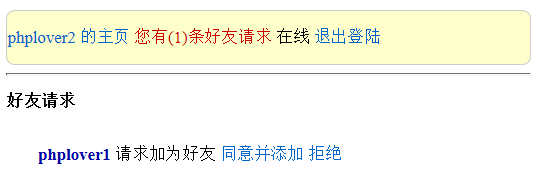


图15 案例结果图（好友请求信息）

⑫点击“同意并添加”的超链接，可以看到如图16所示的添加好友成功的提示信息。

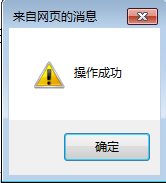


图16 案例结果图（添加好友成功提示信息）

⑬点击【确定】按钮后，点击“返回主页”的超链接，可以看到

用户“phplover2”的好友列表中有了“phplover1，如图17所示”。

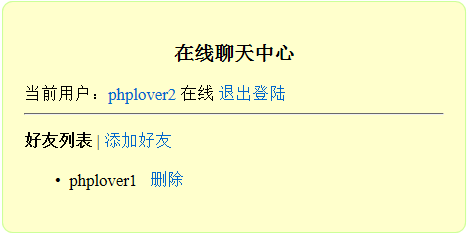


图17 案例结果图（添加好友成功）

⑭此时在用户“phplover1”的聊天主页面的好友列表中也看到有了“phplover2”，如图18所示。



图18 案例结果图（添加好友成功）

⑮此时在用户“phplover1”的聊天主页面中点击好友“phplover2”的超链接，页面跳转到聊天页面，然后输入一条信息比如“hello，你喜欢php吗？”，点击【发送】按钮，可以看到页面如图19所示。

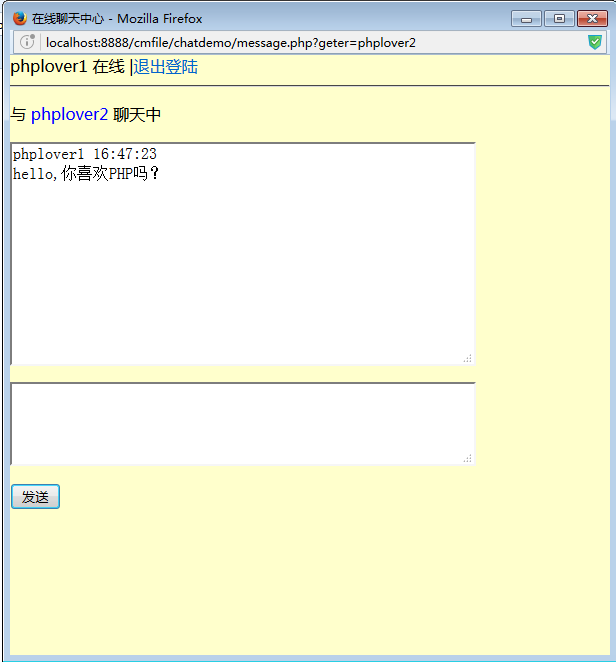


图19 案例结果图（用户“phplover1”向用户“phplover2”发送消息）

⑯此时在用户“phplover2”的聊天页面中也可以看到用户“phplover1”发来的消息，如图20所示。

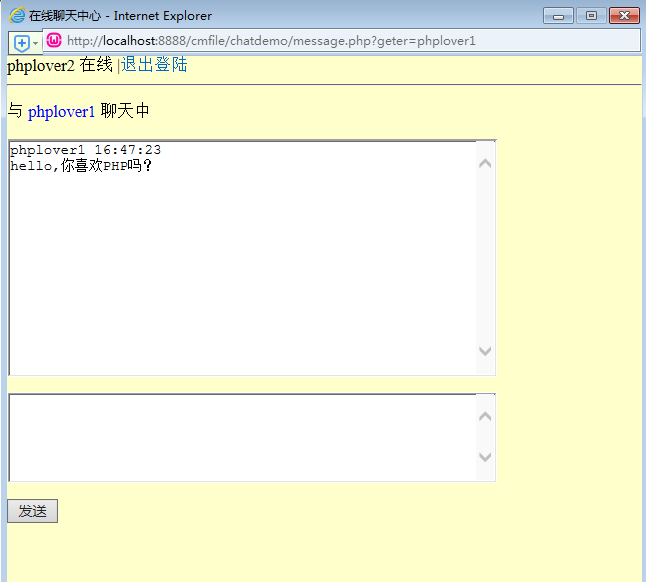


图20 案例结果图（用户“phplover2”成功接收用户“phplover1”发送的消息）

⑰此时用户“phplover2”对用户“phplover1”发来的消息进行回复，比如回复“sure,I think it is a perfect programming language.”，如图21所示。

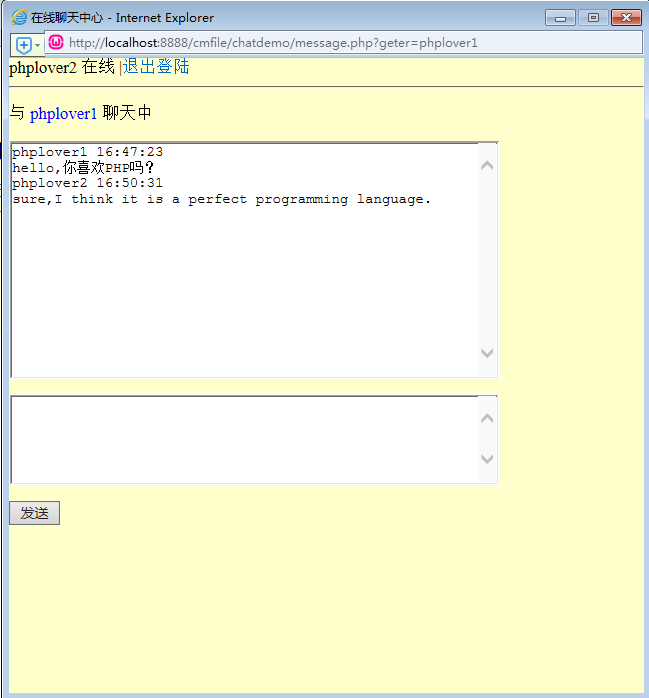


图21 案例结果图（用户“phplover2”对用户“phplover1”发来的消息进行回复）

⑱然后可以看到用户“Phplover1”的聊天页面上也出现了用户“phplover2”发来的消息，如图22所示。

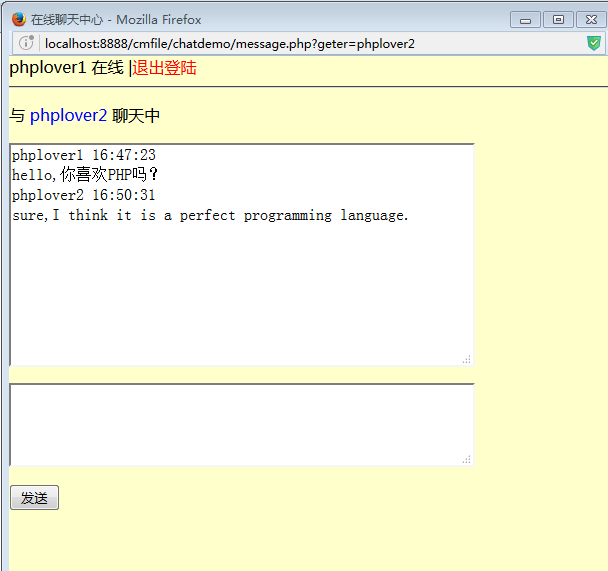


图21 案例结果图（用户“phplover1”成功接收“phplover2”发来的消息回复）

⑲用户“phplover1”关闭聊天页面后，点击好友列表下方好友“phplover2”后面的“删除”超链接，可以看到如图22所示的删除好友成功提示信息。

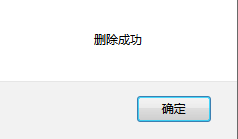


图21 案例结果图（删除好友成功提示信息）

⑳点击【确定】按钮后，可以看到好友列表下方已经没有好友“phplover2”了，如图22所示。



图22 案例结果图（删除好友成功）