**一、简介**

当用户主动发消息给公众号的时候（包括发送信息、点击自定义菜单、订阅事件、扫描二维码事件、支付成功事件、用户维权），微信将会把消息数据推送给开发者，开发者在一段时间内（目前修改为48小时）可以调用客服消息接口，通过POST一个JSON数据包来发送消息给普通用户，在48小时内不限制发送次数。此接口主要用于客服等有人工消息处理环节的功能，方便开发者为用户提供更加优质的服务。

**二、思路分析**

官方文档中只提供了一个发送客服消息的接口，开发者只要POST一个特定的JSON数据包即可实现消息回复。在这里，我们打算做成一个简单的平台，可以记录用户消息，并且用网页表格的形式显示出来，然后可以对消息进行回复操作。

首先，我们使用数据库记录用户主动发送过来的消息，然后再提取出来展示到页面，针对该消息，进行回复。这里我们只讨论文本消息，关于其他类型的消息，大家自行研究。

**三、记录用户消息**

**3.1 创建数据表**

创建一张数据表tbl\_customer 来记录用户消息。

[复制代码](javascript:void(0);)

--

-- 表的结构 `tbl\_customer`

--

CREATE TABLE `tbl\_customer` (

`id` bigint(**20**) unsigned NOT NULL AUTO\_INCREMENT COMMENT '//消息ID',

`from\_user` char(**50**) NOT NULL COMMENT '//消息发送者',

`message` varchar(**200**) NOT NULL COMMENT '//消息体',

`time\_stamp` datetime NOT NULL COMMENT '//消息发送时间',

PRIMARY KEY (`id`),

KEY `from\_user` (`from\_user`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 ;

[复制代码](javascript:void(0);)

**3.2 创建sql.func.php 文件**

创建 \_query($\_sql) {} 函数，来执行INSERT 操作。

[复制代码](javascript:void(0);)

function \_query($\_sql){

if(!$\_result = mysql\_query($\_sql)){

exit('SQL执行失败'.mysql\_error());

}

return $\_result;

}

[复制代码](javascript:void(0);)

**3.3 创建记录消息的函数文件record\_message.func.inc.php**

[复制代码](javascript:void(0);)

//引入数据库处理函数

require\_once 'sql.func.php';

function \_record\_message($fromusername,$keyword,$date\_stamp){

//调用\_query()函数

\_query("INSERT INTO tbl\_customer(from\_user,message,time\_stamp) VALUES('$fromusername','$keyword','$date\_stamp')");

}

[复制代码](javascript:void(0);)

**3.4 处理并记录文本消息**

A. 引入回复文本的函数文件，引入记录消息的函数文件

//引入回复文本的函数文件

require\_once 'responseText.func.inc.php';

//引入记录消息的函数文件

require\_once 'record\_message.func.inc.php';

B. 记录消息入数据库，并返回给用户刚才发送的消息，在这里，你可以修改成其他的文本，比如：“你好，消息已收到，我们会尽快回复您！” 等等。

[复制代码](javascript:void(0);)

//处理文本消息函数

public function handleText($postObj)

{

//获取消息发送者，消息体，时间戳

$fromusername = $postObj->FromUserName;

$keyword = trim($postObj->Content);

$date\_stamp = date('Y-m-d H:i:s');

if(!empty( $keyword ))

{

//调用\_record\_message()函数，记录消息入数据库

\_record\_message($fromusername,$keyword,$date\_stamp);

$contentStr = $keyword;

//调用\_response\_text()函数，回复发送者消息

$resultStr = \_response\_text($postObj,$contentStr);

echo $resultStr;

}else{

echo "Input something...";

}

}

[复制代码](javascript:void(0);)

**四、网页展示用户消息**

我们的最终效果大概如下所示，主要的工作在“信息管理中心”这块，其他的页面布局等等，不在这里赘述了，只讲解消息展示这块。



**4.1 具体实施**

引入数据库操作文件，执行分页模块，执行数据库查询，将查询出来的结果赋给$\_result 供下面使用。

[复制代码](javascript:void(0);)

//引入数据库操作文件

require\_once 'includes/sql.func.php';

//分页模块

global $\_pagesize,$\_pagenum;

\_page("SELECT id FROM tbl\_customer",15); //第一个参数获取总条数，第二个参数，指定每页多少条

$\_result = \_query("SELECT \* FROM tbl\_customer ORDER BY id DESC LIMIT $\_pagenum,$\_pagesize");

[复制代码](javascript:void(0);)

将$\_result 遍历出来，依次插入表格中。

[复制代码](javascript:void(0);)

<form>

<table cellspacing="1">

<tr><th>消息ID</th><th>发送者</th><th>消息体</th><th>消息时间</th><th>操作</th></tr>

<?php

while(!!$\_rows = \_fetch\_array\_list($\_result)){

$\_html = array();

$\_html['id'] = $\_rows['id'];

$\_html['from\_user'] = $\_rows['from\_user'];

$\_html['message'] = $\_rows['message'];

$\_html['time\_stamp'] = $\_rows['time\_stamp'];

?>

<tr><td><?php echo $\_html['id']?></td><td><?php echo $\_html['from\_user']?></td><td><?php echo $\_html['message']?></td><td><?php echo $\_html['time\_stamp']?></td><td><a href="reply.php?fromusername=<?php echo $\_html['from\_user']?>&message=<?php echo $\_html['message']?>"><input type="button" value="回复" /></a></td></tr>

<?php

}

\_free\_result($\_result);

?>

</table>

</form>

[复制代码](javascript:void(0);)

**说明：**在每条消息后，都有一个“回复”操作，点击该按钮，向reply.php文件中传入fromusername和用户发送的消息，为回复用户消息做准备。

**五、消息回复**

**5.1 创建客服消息回复函数文件customer.php**

微信发送客服消息的接口URL如下：

https://api.weixin.qq.com/cgi-bin/message/custom/send?access\_token=ACCESS\_TOKEN

需要POST的JSON数据包格式如下：

[复制代码](javascript:void(0);)

{

"touser":"OPENID",

"msgtype":"text",

"text":

{

"content":"Hello World"

}

}

[复制代码](javascript:void(0);)

所以，根据上面的提示，我们编写处理函数 \_reply\_customer($touser,$content)，调用的时候，传入touser和需要回复的content，即可发送客服消息。

[复制代码](javascript:void(0);)

function \_reply\_customer($touser,$content){

//更换成自己的APPID和APPSECRET

$APPID="wxef78f22f877db4c2";

$APPSECRET="3f3aa6ea961b6284057b8170d50e2048";

$TOKEN\_URL="https://api.weixin.qq.com/cgi-bin/token?grant\_type=client\_credential&appid=".$APPID."&secret=".$APPSECRET;

$json=file\_get\_contents($TOKEN\_URL);

$result=json\_decode($json);

$ACC\_TOKEN=$result->access\_token;

$data = '{

"touser":"'.$touser.'",

"msgtype":"text",

"text":

{

"content":"'.$content.'"

}

}';

$url = "https://api.weixin.qq.com/cgi-bin/message/custom/send?access\_token=".$ACC\_TOKEN;

$result = https\_post($url,$data);

$final = json\_decode($result);

return $final;

}

function https\_post($url,$data)

{

$curl = curl\_init();

curl\_setopt($curl, CURLOPT\_URL, $url);

curl\_setopt($curl, CURLOPT\_SSL\_VERIFYPEER, FALSE);

curl\_setopt($curl, CURLOPT\_SSL\_VERIFYHOST, FALSE);

curl\_setopt($curl, CURLOPT\_POST, 1);

curl\_setopt($curl, CURLOPT\_POSTFIELDS, $data);

curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, 1);

$result = curl\_exec($curl);

if (curl\_errno($curl)) {

return 'Errno'.curl\_error($curl);

}

curl\_close($curl);

return $result;

}

[复制代码](javascript:void(0);)

下面，我们就将上面写好的函数引入到消息回复页面，实现发送客服消息的功能。

**5.2 点击“回复”按钮，带上fromusername和message参数跳转到reply.php。**



**5.3 reply.php 页面显示**



**5.4 reply.php文件分析**

//引入回复消息的函数文件

require\_once '../customer.php';

form表单提交到relpy.php本身，带有action=relpy.

[复制代码](javascript:void(0);)

<form method="post" action="reply.php?action=reply">

<dl>

<dd><strong>收件人：</strong><input type="text" name="tousername" class="text" value="<?php echo $from\_username?>" /></dd>

<dd><strong>原消息：</strong><input type="text" name="message" class="text" value="<?php echo $message?>" /></dd>

<dd><span><strong>内　容：</strong></span><textarea rows="5" cols="34" name="content"></textarea></dd>

<dd><input type="submit" class="submit" value="回复消息" /></dd>

</dl>

</form>

[复制代码](javascript:void(0);)

action=reply 动作处理。

[复制代码](javascript:void(0);)

if($\_GET['action'] == "reply"){

$touser = $\_POST['tousername'];

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

$result = \_reply\_customer($touser, $content);

if($result->errcode == "0"){

\_location('消息回复成功！', 'index.php');

}

}

[复制代码](javascript:void(0);)

**说明：**POST方式获取touser, content，然后调用\_reply\_customer($touser, $content)方法处理，处理成功，则弹出“消息回复成功！”，然后跳转到index.php页面，完成发送客服消息过程。

**六、测试**

**6.1 微信用户发送消息**



**6.2 平台消息管理**

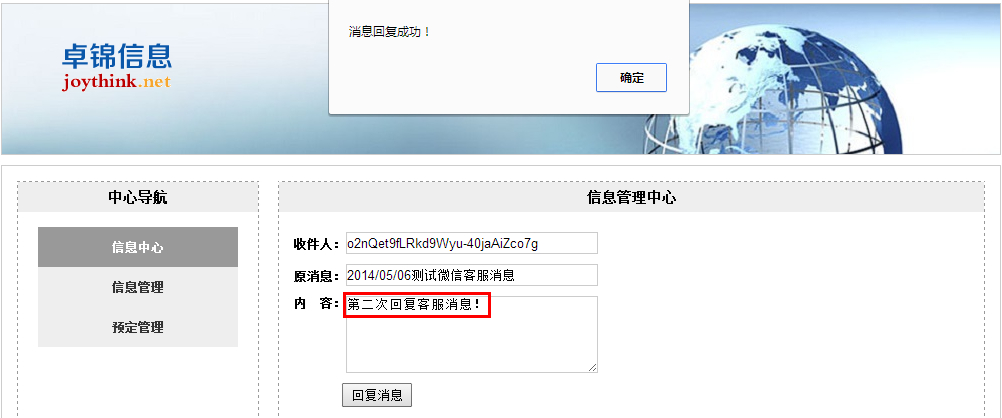


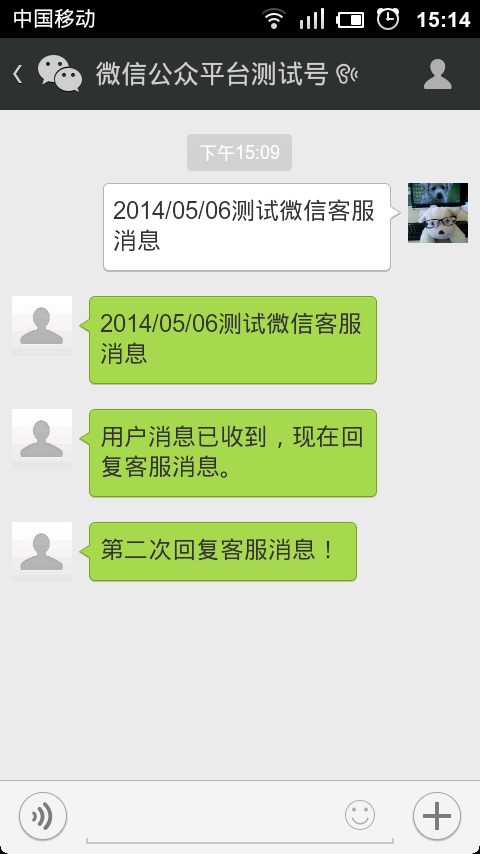
**6.3 发送客服消息**





再次发送客服消息





发送客服消息测试成功！

**七、代码获取**

<http://files.cnblogs.com/mchina/customer.rar>