**青岛农业大学**

**理学与信息科学学院**

**计算机网络综合实训报告**

设 计 题 目 实时聊天室

学生专业班级 计科1900

学生姓名（学号） Paimon Moe（2019000000）

指 导 教 师 Teacher

完 成 时 间 2022年9月17日

设 计 地 点 信息楼

2022年 9 月 23 日

目录

[一、实习任务和目的 1](#_Toc114316972)

[二、任务规划 1](#_Toc114316973)

[三、设计细节 2](#_Toc114316974)

[3.1 客户端设计 2](#_Toc114316975)

[3.2服务器端设计 5](#_Toc114316976)

[四、总结与心得体会 8](#_Toc114316977)

[附录 9](#_Toc114316978)

# 一、实习任务和目的

选题7：实现一个一对多的聊天程序。建议部署在两台电脑上或多台电脑上运行

# 二、任务规划

选题结合笔者网站全栈开发经验，拟采用网站的形式实现一个实时聊天室。网页拥有显示聊天历史消息记录、发送消息的功能，支持高并发，消息实时多端同步。

目标运行环境：Apache服务器端程序、PHP8扩展、Windows 11及以上、Chrome\Chromium\Edge浏览器。

代码编写软件：Visual Studio Code

涉及编程语言：PHP8、JavaScript、HTML5

层叠样式表版本：CSS3

服务端流程图如图2.1所示

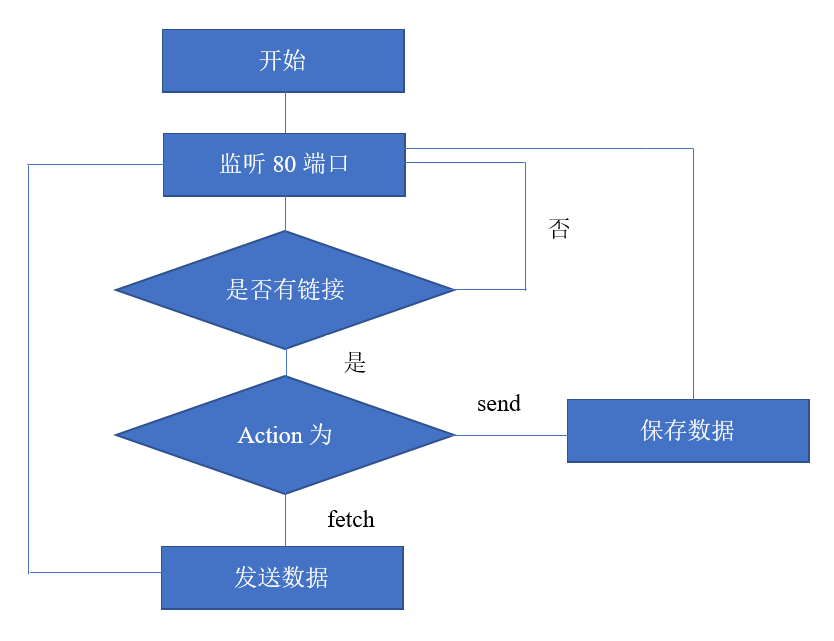


图2. 1

# 三、设计细节

## 3.1 客户端设计

前端设计全局使用字体“造字工房悦黑”，采用现代化设计视觉风格，总览如图 3-1所示。

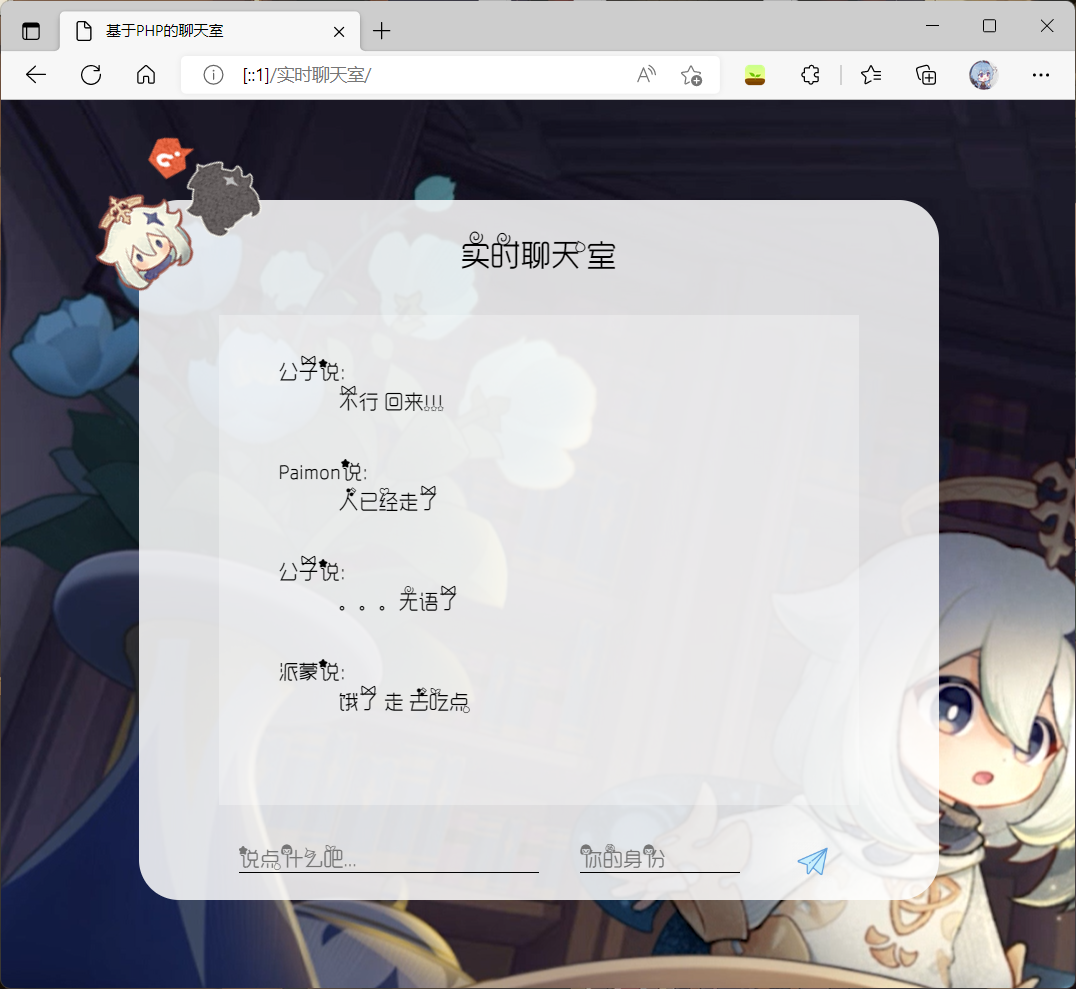


图 3-1

消息实时更新采用JavaScript语言中的Ajax技术，通过创建XMLHttpRequest对象实现，做到了在不刷新页面的情况下更新消息记录的内容。请求方式是额外开辟一个TCP链接，将POST请求发送至服务器监听程序，在请求中携带GET参数“action”，参数值为“fetch”，意为获取消息。该操作每0.5秒进行一次，并将获取的服务端Response在本地字符串分隔后通过DOM操作更新到网页中。涉及代码细节如图 3-2所示。客户端Request与服务端Response报文数据如图 3-3所示。



图 3-2



图 3-3

消息发送功能支持自定义昵称，得益于文本框添加过回车键事件监听器，输入消息之后轻敲空格即可向公共空间发送消息。消息装载于POST请求的Payload中，如图 3-4所示。服务端收到响应后打印出$\_REQUEST变量信息（vardump函数）作为Response，内容如图 3-5所示。公共消息效果如图 3-6所示。

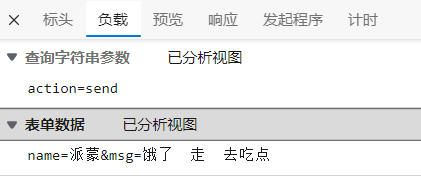


图 3-4

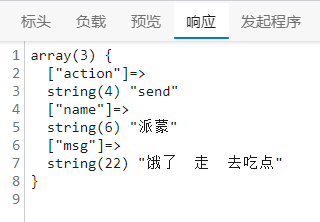


图 3-5

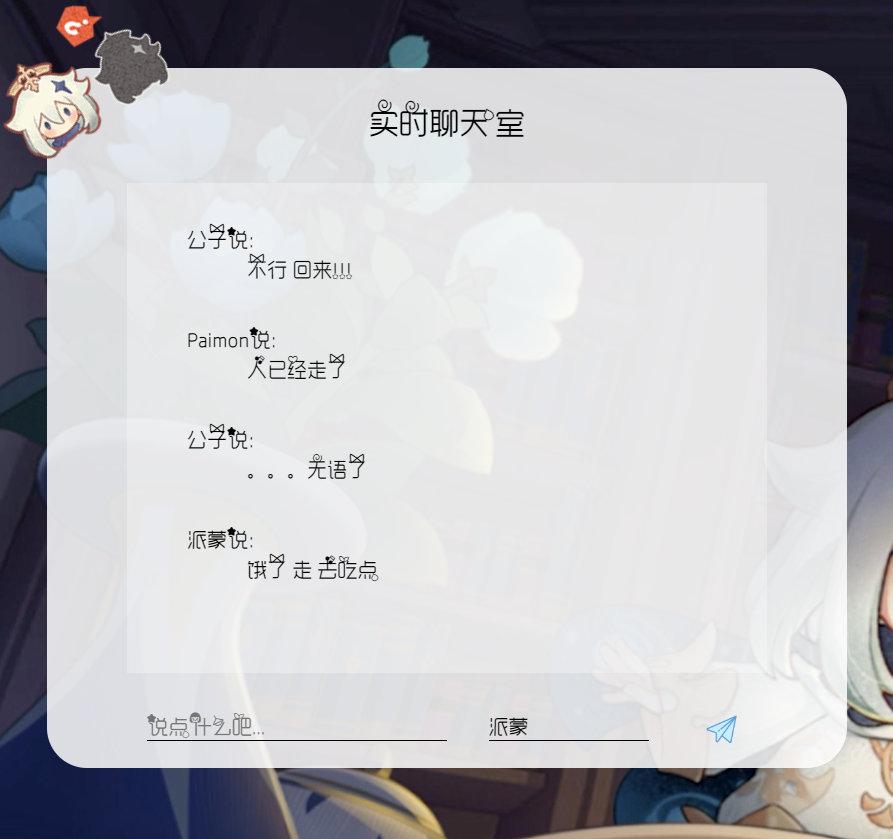


图 3-6

因为只有当POST请求头中包含值为“application/x-www-form-urlencoded”的ContentType时，PHP才会接受POST传值的Payload，所以在使用JavaScript代码模拟该HTTP请求发送的时候需要额外添加这一条目。此时需要传给服务器的GET参数值为“send”，表示要发送消息，请服务器接受并处理，详细代码设计如图 3-7所示。



图 3-7

## 3.2服务器端设计

服务器端使用xampp套件中的Apache程序，将Apache监听端口设置为80（HTTP默认端口），由于本机测试网页无法申请SSL加密证书，所以无需设置443（HTTPS默认端口）端口监听，xampp设置完成之后的控制面板如图 3-8所示。

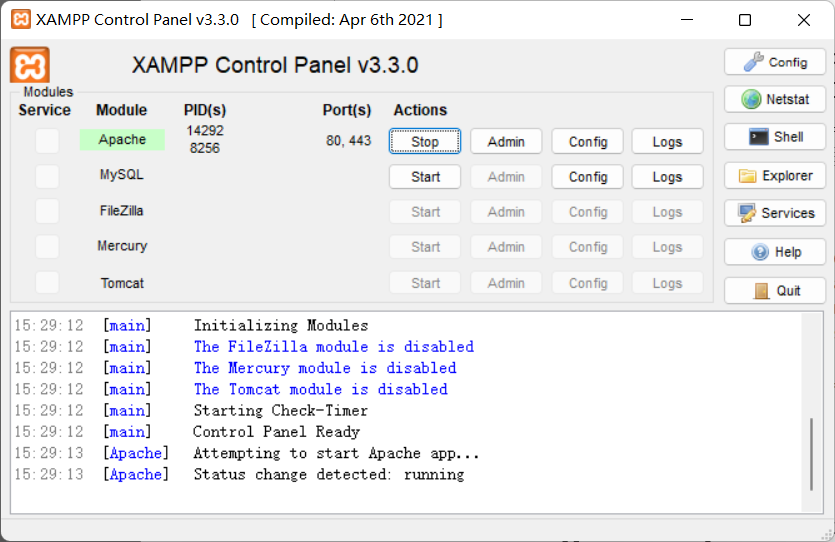


图 3-8

由于聊天信息数据量极低（通常不超过10MB）且无需删除、修改、查找操作，无需使用复杂而繁琐的数据库存储，选择使用小型数据库文件（db文件）保存，可以更有效地节省空间、CPU、时间和磁盘IO，数据的写入读取操作完全依靠PHP原生代码，结合实际需要灵活自定义代码，效率极高，数据库文件如图 3-9所示。

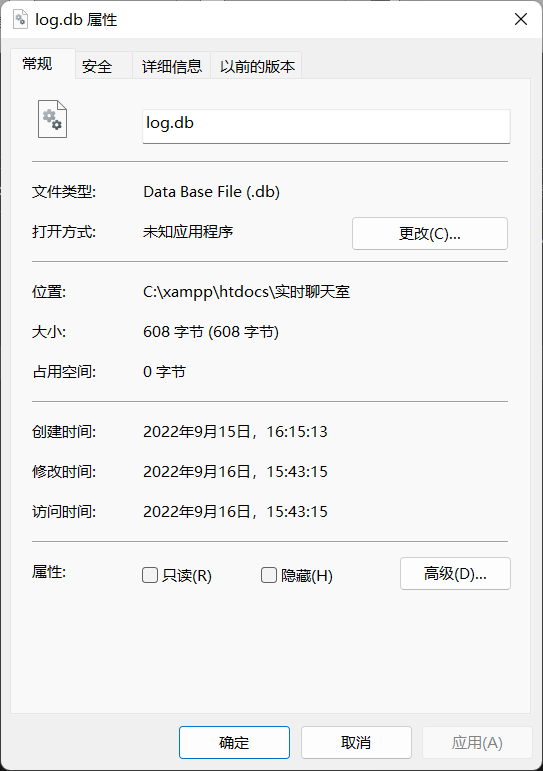


图 3-9

当客户端通过TCP协议发送HTTP请求到服务器80端口时，Apache会立即根据URL定位服务资源所在物理位置（C:\xampp\htdocs\实时聊天室\service.php），并调用PHP插件解析代码。

当客户端发来的消息不包含或包含无法解析的action参数时，将打印出错误提示“\_Error: missing parama 'action' in GET method or unrecognizable val”（\_错误：GET传值方法中丢失’action’参数或参数无法识别）。

当客户端发来的消息中action参数为“fetch”时，执行数据推送操作，将数据库中最新四条消息传回客户端，具体代码细节如图 3-10所示。

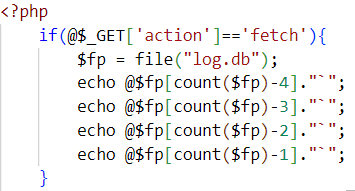


图 3-10

当客户端发来的消息中action参数为“send”时，执行插入消息操作，解析POST传值中的Payload，读取其中的name与msg属性的值，下图以旅行者为昵称，消息为关系太大了为例，展示客户端发送的HTTP报文，如图 3-11所示。



图 3-11

在上图（图3.11）中，第一行表明了HTTP传值方式为POST，又称表单提交，紧接着指明了URL。在URL中，“[::1]”代表本机回环地址，可见服务器就在本机，端口号默认80可以省略，后面是“实时聊天室”五个中文字符的URL编码，“server.php”是服务器代码所在文件名，在“？”后面是GET传值的属性和值。同时header中属性“Content-Type”指明了客户端正在进行有payload的表单提交，提醒服务器接收并解析payload中的信息。最后一行表明该表单传值具有两个属性，其中一个为name，即昵称，另一个为msg，即消息（message）。服务器用于处理该报文的代码如图 3-12所示。

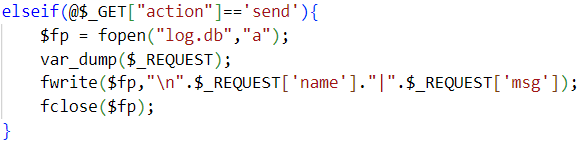


图 3-12

# 四、总结与心得体会

得益于之前的知识与经验，本次实训比较顺利，网站前端设计相比后端逻辑而言耗费了更多的时间（十倍之多），因为笔者认为：对于用户而言，程序的外观设计是第一印象，会严重影响用户的使用体验。为了给潜在的用户一个良好的使用体验，笔者使用了较为前卫的设计风格配合动态背景达到了一个不错的效果，同时字体使用造字工房悦黑体，比标准的黑体、宋体多了许多装饰与艺术元素。本想把网站部署在自己的云服务器上，但是由于带宽限制，客户端完整加载网站需要一分钟左右，遂放弃，改用本地服务器套件。

在消息同步方面，苦于物联网大厂们的技术封锁，无法借鉴使用现代物联网“长连接”模式，遂使用替代方案“轮询”模式，每0.5秒向服务器请求一次最新消息，若以后有更高要求的同步，局域网内通信可将轮询时间改为0.002秒以无限逼近实时，虽会增加服务器与客户端负载，但是现代设备足以应对，低于0.002秒可能会导致不稳定，不建议低于此数值。

通过本次实训使我更加了解了POST与GET传值。GET传值会显示在URL的最后，但POST传值不会，POST在这一点相比于GET传值更加隐蔽。另外，POST还可以传输文件，这是GET传值所不具备的。当然GET也是有优点的，比如用户在将网站分享给他人的时候可以连带GET传值一起复制粘贴，例如在某些视频网站观看视频，分享视频的时候GET传值可以携带一个视频时间数据，可以让他人打开就自动跳转到视频指定位置。

# 附录

|  |
| --- |
| //index.php  <!DOCTYPE html>  <html lang="cn">  <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>基于PHP的聊天室</title>      <style>          @font-face {              font-family: reg;              src: url("res/font\_regular.ttf");          }          \*{              color:rgba(0,0,0,0,.8);              margin: 0px;              padding: 0px;              font-size: 16px;              font-family: reg;              box-sizing: border-box;              transition: .4s;          }          html{              height: 100%;              width: 100%;          }          body {              background-color: black;              position: relative;              text-align: center;              height: 100%;              width: 100%;              overflow: hidden;          }          .bg\_video{              position: absolute;          }          .wrap{              display: inline-block;              background-color: rgba(255,255,255,.85);              height: 35em;              width: 40em;              position: relative;              top:5em;              border-radius: 2em;              text-align:center;          }          .paimon\_icon{              position: absolute;              transform: rotateZ(-36deg);              top:-5em;              left: -3em;            }          .logging{              background-color:rgba(255,255,255,.3);              width: 80%;              height: 70%;              display: inline-block;              position: relative;              top:2em;              padding: 1em;          }          .typewriter{              position: absolute;              bottom: .6em;              left:5em;              /\* background-color: pink; \*/          }          .send{              border: 0;              background-color: rgba(0,0,0,0);              float:right;            }          .item{              padding: 1em 2em 1em 2em;          }      </style>  </head>  <body>      <div class="bg\_video">          <video src="./res/bg\_video.mp4" loop muted autoplay></video>      </div>      <div class="wrap">          <div class="paimon\_icon">              <img src="./res/paimon\_icon.png" alt="png\_load\_error">          </div>          <div style="font-size: 1.5em;margin-top: 1em;">              实时聊天室          </div>          <div class="logging">              <div id="log0" class="item">                  <div id="log0\_name" style="text-align:left;">                    </div>                  <div id="log0\_msg" style="text-align:left;padding-left: 3em;">                    </div>              </div>              <div id="log1" class="item">                  <div id="log1\_name" style="text-align:left;">                    </div>                  <div id="log1\_msg" style="text-align:left;padding-left: 3em;">                    </div>              </div>              <div id="log2" class="item">                  <div id="log2\_name" style="text-align:left;">                    </div>                  <div id="log2\_msg" style="text-align:left;padding-left: 3em;">                    </div>              </div>              <div id="log3" class="item">                  <div id="log3\_name" style="text-align:left;">                    </div>                  <div id="log3\_msg" style="text-align:left;padding-left: 3em;">                    </div>              </div>            </div>          <div class="typewriter">              <input type="text" placeholder="说点什么吧..." id="input\_msg" onkeydown="if(event.keyCode==13){xhr\_msg\_send()}" style="background-color:rgba(0,0,0,0);border: none;border-bottom:1px solid black;outline:none;width:15em;">              &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;              <input type="text" placeholder="你的身份" id="input\_name" style="background-color:rgba(0,0,0,0);border: none;border-bottom:1px solid black;outline:none;width:8em;">              &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;              <button class="send" id="btn\_send" onclick="xhr\_msg\_send()">                  <img src="./res/send\_icon.png" alt="pnd\_load\_error" style="width:2em;">              </button>          </div>      </div>      <script>            function xhr\_msg\_fetch(){              var xhr\_fetch = new XMLHttpRequest();              xhr\_fetch.open("POST","service.php?action=fetch");              xhr\_fetch.send();              xhr\_fetch.onreadystatechange = function(){                  if(xhr\_fetch.readyState == 4 && xhr\_fetch.status == 200){                      // alert(xhr\_fetch.responseText);                      var fetch\_str = xhr\_fetch.responseText;                      var array = fetch\_str.split("`");                      for(var i=0;i<4;i++){                          var name = array[i].split("|")[0];                          var msg = array[i].split("|")[1];                          document.getElementById("log"+i+"\_name").innerHTML=name+"说：";                          document.getElementById("log"+i+"\_msg").innerHTML=msg;                      }                  }              };          }          var msg\_fetch\_timer = setInterval("xhr\_msg\_fetch()",500);          function xhr\_msg\_send(){              var xhr\_send = new XMLHttpRequest();              xhr\_send.open("POST","service.php?action=send");              xhr\_send.setRequestHeader("content-type","application/x-www-form-urlencoded");              var name = document.getElementById("input\_name").value;              var msg = document.getElementById("input\_msg").value;              xhr\_send.send("name="+name+"&msg="+msg);              document.getElementById("input\_msg").value="";          }      </script>  </body>  </html>  //service.php  <?php      if(@$\_GET['action']=='fetch'){          $fp = file("log.db");          echo @$fp[count($fp)-4]."`";          echo @$fp[count($fp)-3]."`";          echo @$fp[count($fp)-2]."`";          echo @$fp[count($fp)-1]."`";      }      elseif(@$\_GET["action"]=='send'){          $fp = fopen("log.db","a");          var\_dump($\_REQUEST);          fwrite($fp,"\n".$\_REQUEST['name']."|".$\_REQUEST['msg']);          fclose($fp);      }      else{         //phpinfo();          echo "\_Error: missing parama 'action' in GET method or unrecognizable val";      }  ?> |

**计算机网络综合实训成绩评定表**

|  |  |  |  |
| --- | --- | --- | --- |
| 学生姓名 |  | 专业班级 |  |
| 设计题目 |  | | |
| 指导教师评语及意见：  指导教师评阅成绩： 指导教师签字：    年 月 日 | | | |

注：此表装订在报告之后。