

基础实验4：聊天墙

1. 实验背景

Lab μ 是清华大学开发校园应用的社团，近日，CC 同学开发了微信墙的后台，希望在校内各种活动和许多学生节使用，可是他对前端不太熟悉，你能帮帮他吗？

学以致用，本次作业，让我们借助 websocket 和 CSS3，来实现一个简单的聊天墙吧。

2. 实验要求

2.1 基础要求

- 能够正确地在页面上显示3-5条消息（5分）
- 正确显示用户昵称和头像（5分）
- 考虑到用户的头像加载可能较慢，做一些合适的加载动画处理（5分）
- 能够在接受新消息时将消息动态显示在页面上，要求删除之前的消息（10分）
- 在合适的位置实现滚动的伪公告（10分）
- 在刷新页面时能够加载历史消息并显示（5分）
- 适配不同的电脑屏幕尺寸，要求页面总能占满全屏且布局合理（在PC端正常工作即可，不用考虑移动端显示效果）（5分）
- 基本的 UI 设计（30分）
 - 设计一套符合大众审美，至少是新世纪风格的UI
 - 考虑使用场景（大型活动等），保证字号不过小
 - 消息长度不超过100个字，请选择合适的方式展示，例如 消息过长可以滚动 等
 - 考虑长度完全不同的消息出现在统一屏幕上的合适展示效果，例如 统一字体，垂直居中等（仅供参考）
- 使用 CSS3 动画，让新消息加入和旧消息删除的过程更加流畅（5分）
- 实现管理员账号的消息显示（20分）
 - 和基本消息的样式不同
 - 在页面上作为最新消息置顶十秒，且不影响别的消息的接收
 - 如果十秒内收到另一条新的管理员消息，覆盖并重新计时
 - 注意代码复用

参考样例：



注：你可以用任何其他排版布局，只要实现以上功能就好。

2.2 提高要求

- 实现正确显示 emoji 和微信表情，可以使用第三方库（[twemoji](#), [weface](#) 等）（加分不超过 3 分）
- 使用数据绑定模型（Vue.js、Angular.js 等）实现前端页面的数据绑定（加分不超过 5 分）
- 一些特殊的适用于聊天墙/微信墙的其他元素（加分不超过 2 分）
- 加分总和不超过10分

2.3 其他说明

- 服务器端发送的消息已经做过 HTML 转义处理，避免有人 XSS 注入时导致页面卡死。
- 可以使用第三方库如 jQuery，外部样式与动画效果等。
- 不能成套引用外部插件。基本的界面与事件响应需自己实现，否则扣除该部分分数。
- 需要在文档中简单说明实现的功能与相关技术。
- 需要在文档中区分自己所做的工作和对他人成果的引用。

2.4 注意事项

- 不得抄袭。抄袭与被抄袭双方，本次作业按 0 分处理。
- 禁止利用 API 对服务器恶意攻击。
- 按照作业提交一节中的要求，进行提交与发布。
- 鼓励使用较新的技术，网页需要至少支持最新的 Chrome 浏览器。

3. 接口说明

后台地址：<https://wall.cgcbcbc.com>

微信号：secourse



可以向软件工程的微信号发送消息，消息会在后台通过 WebSocket 发送给所有客户端（即你们的浏览器），使用 socket 连接到 后台地址，并监听 `new message` 事件即可获取普通用户的新消息。

消息格式：

```
1 {  
2   "nickname": "小\z",  
3   "content": "测试消息",  
4   "headimgurl": "http://xxx.xxx.xx/xxx"  
5 }
```

你可以通过向微信号发送「admin 管理员消息内容」伪装管理员发送消息，消息会在后台通过 WebSocket 发给所有客户端，使用 socket 连接到后台，并监听 `admin` 事件即可获得管理员消息。

消息格式：

```
1 {  
2   "nickname": "admin",  
3   "content": "测试管理员消息"  
4 }
```

你可以自己选择管理员头像放在本地。

你也可以通过 `ajax` 批量获取历史消息，通过 `get` 请求 `/api/messages?num=3`

参数 num 为获取的消息条数，默认为3，按发送消息的时间排序。

```
1  [
2    {
3      "nickname": "CC",
4      "content": "测试消息",
5      "headimgurl": "http://xxx.xxx.xx/xxx"
6    },
7    {
8      "nickname": "小T",
9      "content": "测试消息2",
10     "headimgurl": "http://xxx.xxx.xx/xxx"
11   },
12   {
13     "nickname": "小Z",
14     "content": "测试消息3",
15     "headimgurl": "http://xxx.xxx.xx/xxx"
16   }
17 ]
```

4. 作业提交

请在网络学堂提交一个 zip 或 rar 格式的压缩包，文件名为「学号_姓名_作业次数」，压缩包中包括本次作业的源代码、说明文件及其他必要的文件。其中源代码请放入压缩包根目录中的 `src` 文件夹；说明文件只接受 pdf、md 或 html 格式，请命名为 report.pdf、report.md 或 report.html，在文档中请务必写明姓名、学号、联系邮箱；其他必要文件请放入压缩包根目录中的 ext 文件夹。

请在 Github Pages 中发布本次实验成果，在文档和课程作业提交内容中都写明发布的链接。