内部文档V1

暂定有四个页面（webpage），分别是welcome页面、home页面、帖子页面和用户页面。

welcome页面是打开网页后的内容（相当于bbs的登录页面），包含一个大logo，一个用户id输入框，一个用户密码输入框，一个注册按钮，一个登录按钮，一个游客按钮。

点击注册按钮后，浏览器向服务器发送一个json字符串，格式是{‘id’: ‘xxx’, ‘pwd’:’xxx’}，后期实现可以用RSA算法对id和pwd进行加密。然后会刷新页面，等待服务器返回一个json字符串表示注册成功或失败。格式为{‘status’: 0/1}，0表示成功，1表示失败。

点击登录按钮后，浏览器发送和上面一样的json字符串，并等待服务器返回一个json字符串表示登录成功或失败，格式和意义和上面一样。登录成功则自动跳转到home页面。

点击游客按钮后，浏览器发送json字符串，格式为{‘mac’:’xxxxx’}发送mac地址，然后登录成功进入home页面。

注意：后两个按钮发送服务器数据后需要服务器建立一个会话并且返回一个cookie标识。

home页面，页面最上方，左边是logo，右边是用户名字，点击可以进入用户页面。下面的主体是按照某种规则排序的十条帖子。这十条帖子是浏览器通过get请求向服务器获得的。请求的参数为page=1（表示第一页，前十条），具体请求可以是xxx(域名)/home?page=1

然后服务器返回一个json字符串，格式为

{

‘status’: 0/1,

‘data’:[

{

‘title’:’xxx’,

‘num’: 111,

‘time’: ‘xxx’,

‘last’:’xxxx’,

‘id’: ‘xxx’,

‘tid’:’xxx’

}

]

}

status表示状态，0表示无误，1表示有问题；data是个数组，数组的每个元素含有title表示帖子标题、num表示跟帖数量、time表示发帖时间、last表示最后回帖时间、id表示发帖人id、tid表示帖子id。

这个页面最下方还有一个发布新帖的文本框和选项。发送新帖后浏览器会给服务器发送一个json字符串，格式是

{

‘title’:xxx,

‘text’:xxx,

‘id’:xxx,

‘time’:xxx

}

分别是帖子标题，发帖内容，发帖人id，发帖时间。

用户页面，页面最上方和home一样。主要的内容是显示用户的发帖纪录。同样靠浏览器发送请求实现，具体请求可以是xxx(域名)/user?id=xxxx&page=1

服务器返回一个json字符串，内容和上面的大同小异，其中data数组的元素需要改成

{

‘title’:xxx,

‘text’:xxx,

‘tid’:xxx,

‘time’:xxx,

}

title表示在这个标题的帖子下发帖，text表示发送内容，tid表示帖子id，time表示发送时间。

帖子页面，页面最上方（同上）。然后次上方是帖子标题。显示各个帖子。需要解析alice、bob等内容。还是通过浏览器的请求，格式为xxx(域名)/thread?tid=xxx&page=1

服务器返回一个json字符串，格式大同小异，其中data数组的元素改成

{

‘text’:xxx,

‘time’:xxx,

‘id’:xxx,

‘fid’:xxx

}

意思分别是，某一楼的内容，发帖时间，发帖人id，楼id。

这个页面还要实现一个对某一楼的回帖功能（或者不用，直接1楼、2楼、3楼，没有回复关系），发送给服务器的json数据格式为

{

‘fid’:xxx,

‘id’:xxx,

‘text’:xxx,

‘time’:xxx

}

意思分别是回复某一楼的id、用户id、回复文本、回复时间。

最后说一句：帖子id和楼id是分开编码的，而且均是全站唯一（可以按照发帖顺序编码）。比如帖子1的第一楼的楼id是1，二楼的楼id是2。然后有人发了帖子2，则帖子2的第一楼的id就是3。