|  |  |
| --- | --- |
| **输入参数** | **描述** |
| openid | 用户在微吧商户的openidID |

## 公众号微喵上墙模块：

**上大屏幕-最后访问酒吧显示**

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| openid | 用户在微吧商户的openidID |
| User\_id | 用户ID |
| merchants\_name | 商家名称 |
| producerlogo\_url | 商户log链接地址 |
| create\_time | 创建时间 |



**上大屏幕-功能列表：**

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 微吧商户ID |
| openidID | 用户在微吧商户的openidID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 微吧商户ID |
| openidID | 用户在微吧商户的openidID |
| size | 记录行数 |
| data | 列表 |

示范json:

{

"merchantid": "10000",

"openidID": "okkdfr",

"size": "5",

"data": [

{

"gameid": "1",

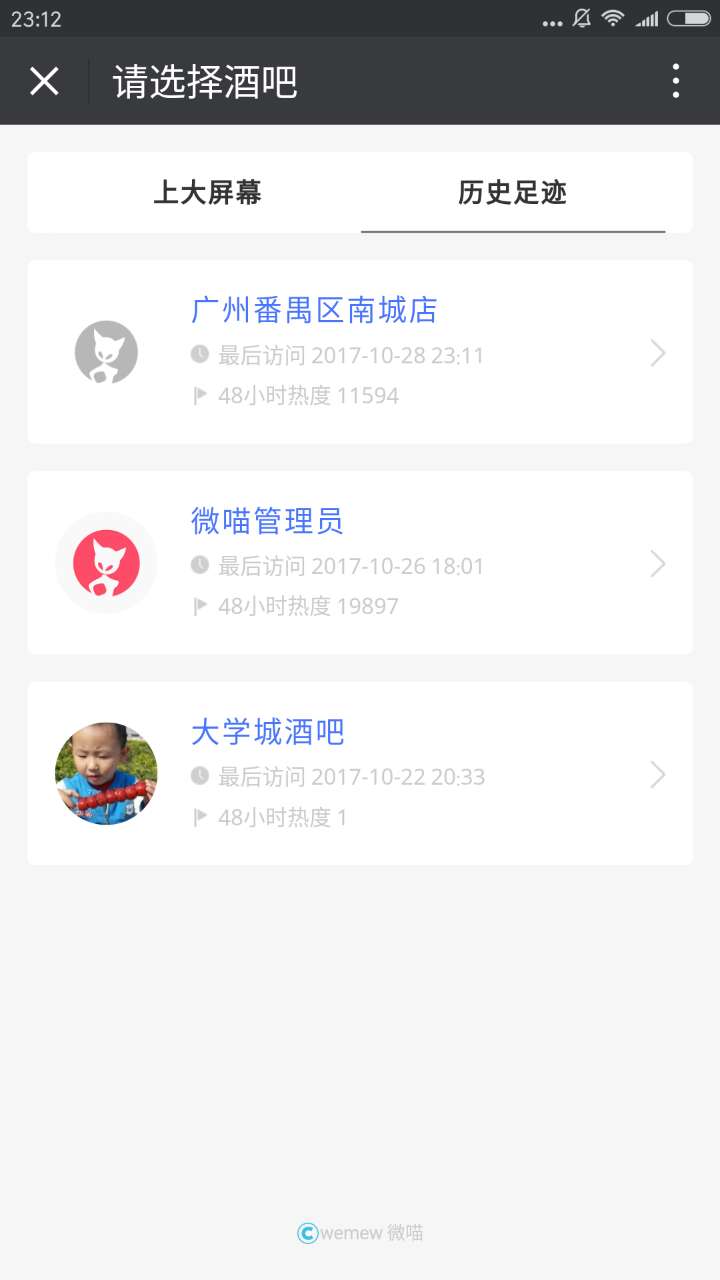
"gamename": "赛马"

}

]

}

**历史足迹-历史足迹列表**



|  |  |
| --- | --- |
| **输入参数** | **描述** |
| openid | 用户在微吧商户的openidID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| openid | 用户在微吧商户的openidID |
| User\_id | 用户ID |
| Data | 列表信息 |

示范json:

{

"merchantid": "10000",

"openidID": "okkdfr",

"size": "5",

"data": [

{

"merchants\_id": "1", //商户ID

"merchants\_name": "1", //商户名称

"producerlogo\_url": "whh.jpg",// 商户图片

"create\_time": "2017-05-04"// 商户图片

}

]

}

## 公众号红包提现模块：



**红包提现-可提现余额查询**

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| openid | 用户在微吧商户的openidID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| openid | 用户在微吧商户的openidID |
| User\_id | 用户ID |
| balance | 用户当前余额 |

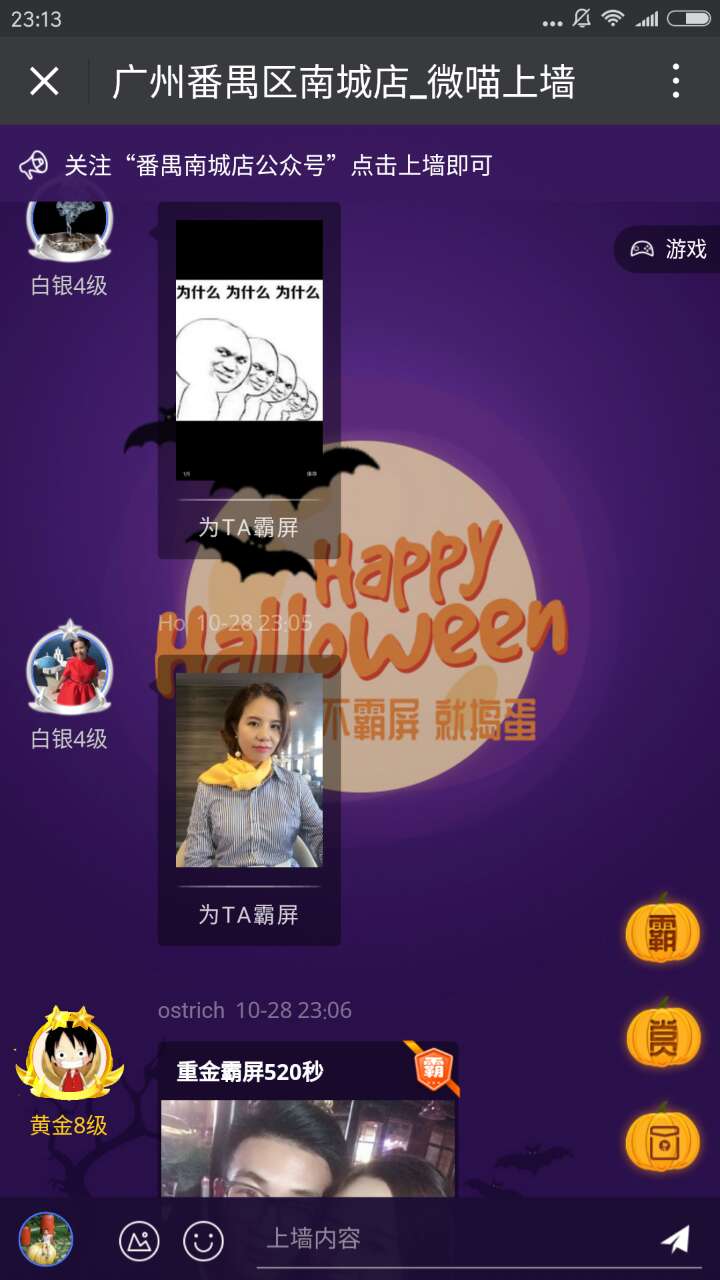
**红包提现-提现操作**

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| openid | 用户在微吧商户的openidID |
| amount | 提现金额 |
| type | 提现方式 0 微信 1支付宝 2银行卡 |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| openid | 用户在微吧商户的openidID |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

## H5移动模块：微喵上墙

**聊天室信息列表： 进入聊天室显示列表，点击用户头像，能够出现第二个效果页**





|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| begtime | 开始时间 |
| endtime | 结束 |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| data | 返回列表信息 |

示范json:

{

“code”:0,

“msg”:”成功”,

"merchantid": "10000",

" uuid ": "10000",

"openidID": "okkdfr",

"size": "5",

"data": [

{

" message\_id": "1", //发送消息id

" user\_id": "1", //发送消息用户id

" user\_level": "1", //用户等级

" nickname": "ds", //用户昵称

" user\_picture ": "1.jpg", //发送消息用户图像

"sex ": "1", //用户性别

" context": "13333", //用户消息内容HTML

" type": "0", //0 文字消息，1 图文消息（含图片）该状态可用霸频 2 打赏消息 3 红包消息 4 点歌 5 投票消息 999其他

"status": "1",// 是否允许霸频,霸过后不允许重复 0,未霸频 1 已霸频

// 如果是红包消息 0 未抢完（前端显示拆红包） 1 已抢完

" redpackageid ": "13333", //红包组ID，红包消息此值不能为空

" screen\_num ": "13333", //霸屏次数，存在连送的情况

"is\_self": "1",//0是 1不是 是否自己发送的消息（自己发送的可以删除）

"create\_time": "2017-05-04"// 消息发送时间

}

]

}

**已霸屏（支持图文，图，文字）：**



**未霸屏（支持图文，图，文字）：**



**打赏消息：（支持图文，图，文字）：**



**可抢红包/抢完红包 两种情况**



**普通图文消息：**



**重点关注:前端需要考虑H5端:普通图文消息，可抢红包,抢完红包（抢到/未抢到）,已霸屏,未霸屏,打赏消息 这6种消息展示情况。**

**用户自己删除聊天室信息： 提供用户自己删除发送消息**

****

删除用户聊天信息weibar\_merchants\_barrage\_info where id=’’ and openid=’’ and merchantid=’’ ;

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| openid | 当前用户在微吧的openid |
| message\_id | 消息ID |
| context | 发送消息的HTML内容 |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

**聊天室发送消息： 提供消息发送接口（仅为图文）**



该功能主要实现将用户聊天信息写入到weibar\_merchants\_barrage\_info

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| type | 0图文消息（含图片）该状态可用霸频，1霸频消息 2 打赏消息 3 红包消息 4 点歌 5 投票消息 999其他 |
| context | 发送消息的HTML内容 |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

**聊天室发送红包消息：  提供红包发送，装入红包能力**

****

1.写入： weibar\_redpackage\_deposit\_temp

2.唤起微信SDK，进行支付

3，微信支付成功回调（异步），写入weibar\_redpackage\_deposit，并对红包个数随机金额拆单批量插入weibar\_redpackage\_deposit\_detail （未领取前openid,领取时间等为空）

4写入PC屏幕特效任务表special\_effects\_lists\_task

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| type | 0图文消息（含图片）该状态可用霸频，1霸频消息 2 打赏消息 3 红包消息 4 点歌 5 投票消息 999其他 |
| rpamount | 红包总金额 |
| rpnumeber | 红包总个数 |
| rpremark | 红包留言 |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

**聊天室抢红包接口：  提供抢红包接口**



1. 前端显示拆红包UI后，点击抢购
2. 进行拆的动作后调用此接口，根据redpackageid 查询 weibar\_redpackage\_deposit 已经发生红包sendnumber大于 领取的红包getnumber。
3. 如果发送红包大于领取的红包，则更新 weibar\_redpackage\_deposit\_detail 设置领取人

Openid，领取时间get\_time，

1. 更新主表getnumber领取数量，如果当前领取数据=发放数量，则更新主表weibar\_redpackage\_deposit的状态为已领取完毕。
2. 异步线程同步调用微信红包接口，并写入weibar\_red\_packet\_order（暂时不做队列异步调用）
3. 写入PC屏幕特效任务表special\_effects\_lists\_task

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| type | 0图文消息（含图片）该状态可用霸频，1霸频消息 2 打赏消息 3 红包消息 4 点歌 5 投票消息 999其他 |
| redpackageid | 红包组ID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| amount | 红包抢购成功金额 |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

**聊天室抢红包接口：  提供查询接口-看看大家手气功能**

查询 weibar\_redpackage\_deposit\_detail 红包领取明细 和 用户基础信息表 user\_base\_info



|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| type | 0图文消息（含图片）该状态可用霸频，1霸频消息 2 打赏消息 3 红包消息 4 点歌 5 投票消息 999其他 |
| redpackageid | 红包组ID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| nickname | 用户的昵称 |
| amount | 红包抢购成功金额/未抢购成功为0 |
| rpamount | 红包总金额 |
| rpnumeber | 红包总个数 |
| rpremark | 红包留言 |
| data | data为json格式列表 |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

"data": [

{

" openid": "1", //发送消息用户id

" nickname": "ds", //用户昵称

" user\_picture ": "1.jpg", //发送消息用户图像

"is\_best": "1", //是否手气最佳

" amount": "13333", //金额

"get\_time": "2017-05-04"// 消息发送时间

}

]

**聊天室发送霸屏信息：（该操作为酒吧,平台分成）**



1. 点击霸屏图标，弹出霸屏DIV层
2. 先使用用户余额查询接口 当用户余额不够的时候前端打开微信的jssdk调用微信的下单接口，写入user\_recharge\_order ，完成支付通知回调更新user\_recharge\_order订单状态。写于用户钱包交易流水平衡信息表user\_wallet\_balance\_log
3. 加用户余额 user\_account\_balance，再对余额进行组合扣费。
4. 扣费成功写入user\_wallet\_balance\_log
5. 如果余额足则直接扣费。

6． 写入PC屏幕特效任务表special\_effects\_lists\_task

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| type | 0图文消息（含图片）该状态可用霸频，1霸频消息 2 打赏消息 3 红包消息 4 点歌 5 投票消息 999其他 |
| goods\_id | 购买霸屏/打赏商品ID |
| goods\_name | 霸屏商品 |
| snum | 霸屏次数 |
| amount | 霸屏总金额 |
| sytle | 霸屏主题 |
| context | 图文html信息 |
| messageid | 为他人霸屏的消息ID |
| beneficial\_user\_id | 如为他人霸屏的消息ID发送者的用户ID。自己霸屏则为个人ID |
| beneficial\_openid | 为他人霸屏的消息ID发送者的微信用户ID。自己霸屏则为个人ID |
| mode | 1 自己霸屏  2 为他人霸屏  3. 打赏个人  4.打赏艺人 |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

**聊天室发送打赏信息：（该操作为酒吧,平台分成。如果打赏用户，用户也需要分成。艺人ID通过后台进行配置）**



1. 点击打赏图标，弹出霸屏DIV层
2. 先使用用户余额查询接口 当用户余额不够的时候前端打开微信的jssdk调用微信的下单接口，写入user\_recharge\_order ，完成支付通知回调更新user\_recharge\_order订单状态。写于用户钱包交易流水平衡信息表user\_wallet\_balance\_log
3. 加用户余额 user\_account\_balance，再对余额进行组合扣费。
4. 扣费成功写入user\_wallet\_balance\_log
5. 如果余额足则直接扣费。

6． 写入PC屏幕特效任务表special\_effects\_lists\_task

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| type | 0图文消息（含图片）该状态可用霸频，1霸频消息 2 打赏消息 3 红包消息 4 点歌 5 投票消息 999其他 |
| goods\_id | 购买打赏商品ID |
| goods\_name | 打赏商品 |
| snum | 打赏次数 |
| sytle | 打赏主题 |
| context | 图文html信息 |
| beneficial\_user\_id | 为他人/艺人的消息ID发送者的用户ID |
| beneficial\_openid | 为他人/艺人打赏的消息ID发送者的微信用户ID |
| mode | 1 自己霸屏  2 为他人霸屏  3. 打赏个人  4.打赏艺人 |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| openid | 当前用户在微吧的openid |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

**聊天室特效处理：前端轮询特效任务（后期性能问题改成队列）**

1. 前端js轮询调用查询接口,获取最早的5条待推送PC端的特效
2. 通过merchantid查询special\_effects\_lists\_task

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |
| size | 当前用户在微吧的openid |
| data | json的list信息 |

"data": [

{

" id": "1", //ID

" openid": "1", //发送消息用户id

" user\_id ": "1", //发送消息用户id

" nickname": "ds", //用户昵称

" user\_picture ": "1.jpg", //用户图像

" pay\_type": "ds", //购买商品类型 1 自己霸屏 2 为他人霸屏 3. 打赏个人 4.打赏艺人 5 红包特效

" beneficial\_user\_id": "ds", //受益用户ID，为他人霸屏，打赏目标

" beneficial\_openid": "ds", //受益者的微信openid'

" beneficial\_user\_picture ": "1.jpg", //发送消息用户图像

" goods\_name": "ds", //打赏商品名称

" goods\_amount": "ds", //打赏商品金额

" goods\_num": "ds", //打赏次数

" sreen\_name ": "1.jpg", //霸屏商品名称

" sreen\_times": "1", //霸屏时长

" sreen\_num ": "13333", //霸屏次数

" redpackage\_amount": "1.jpg", //红包金额

" redpackage\_num": "1", //红包个数

" tip\_template\_tpye": "13333", //打赏效果模板类型1.默认 2.女神 3.土豪 " template\_tpye ": "11"//模板类型1.默认 2.女神 3.土豪

}

]

**聊天室特效处理：特效任务执行完毕后回调更新状态（后期性能问题改成队列）**

1. 通过merchantid和 special\_effectsid查询special\_effects\_lists\_task（对应ID字段）进行特效任务的删除。

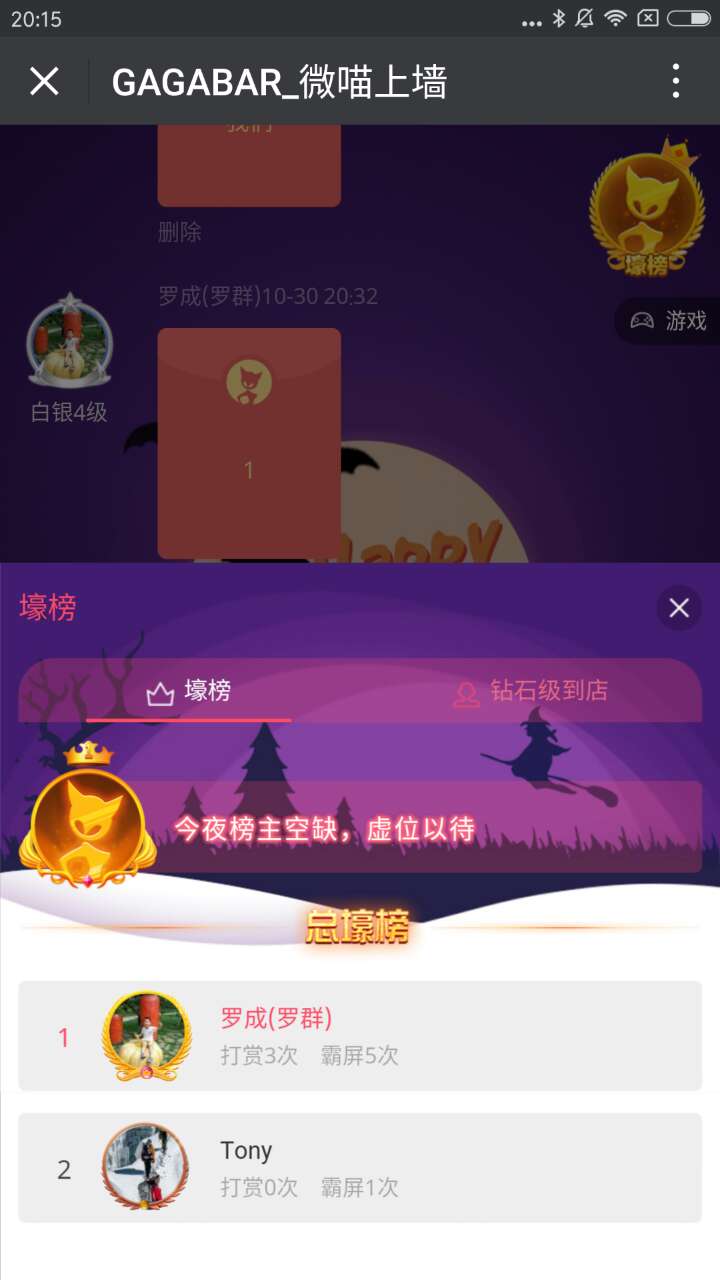
|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |
| special\_effectsid | 特效任务ID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |

**聊天室榜单：土豪榜（今夜榜主）**

土豪排行日榜基本信息表（中午12~第二天中午12点）

根据商户ID查询 plutocrat\_ranking

****

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |
| size | 记录行数 |
| date | 排行榜日期 |
| data | json的list信息 |

"data": [

{

" id": "1", // ID

" user\_id ": "1", //发送消息用户id

" nickname": "ds", //用户昵称

" user\_picture ": "1.jpg", //用户图像

" openid": "1", //发送消息用户id

" is\_best ": "1", //是否最土豪

" amount ": "1", //打赏霸屏金额

}

]

**聊天室榜单：（大R）钻石到店**

****

通过商户ID查询 plutocrat\_user\_info 关联 weibar\_merchants\_login\_log。查看那些大R是钻石到店。

|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |
| size | 记录行数 |
| date | 当前日期 |
| data | json的list信息 |

"data": [

{

" id": "1", // ID

" user\_id ": "1", //发送消息用户id

" nickname": "ds", //用户昵称

" user\_picture ": "1.jpg", //用户图像

" openid": "1", //发送消息用户id

" amount ": "1", //打赏霸屏金额

}

]

**聊天室榜单：用户私聊 （是否在公众号加聊天好友信息）**

**weibar\_private\_chat**



|  |  |
| --- | --- |
| **输入参数** | **描述** |
| begintime | 开始时间 |
| endtime | 结束时间 |
| from\_useropenid | 当前用户自身openid |
| to\_user\_openid | 聊天用户的openid |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |
| from\_useropenid | 当前用户自身openid |
| to\_user\_openid | 聊天用户的openid |
| nickname | 用户昵称 |
| user\_picture | 用户图像 |
| to\_nickname | 用户昵称 |
| to\_user\_picture | 用户图像 |
| size | 记录行数 |
| data | json的list信息 |

"data": [

{

" id": "1", // ID

"message": "1", //消息内容

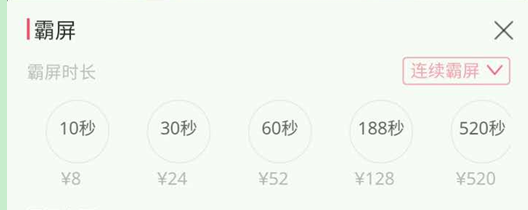
" status": "1"

}

]

**聊天室霸屏时长列表：该商户配置的霸屏商品价格**

根据商户ID 查询**weibar\_goods**返回列表



|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |
| size | 记录行数 |
| data | json的list信息 |

"data": [

{

" id": "1", // ID

" goods\_code": "1", //礼物ID

" goods": "1", //礼物名称

" amount": "ds"

}

]

**聊天室打赏商品列表：该商户配置的打赏商品价格**

根据商户ID 查询weibar\_merchants\_goods\_setting\_info返回列表



|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |
| size | 记录行数 |
| data | json的list信息 |

"data": [

{

" id": "1", //特效任务ID

" goodsid": "1", //礼物ID

" goodsname": "1", //礼物名称

" goodsurl": "ds", //礼物图片

" amount": "1.jpg"

}

]

**聊天室艺人列表：该商户配置的酒吧艺人角色(可以从后台将关注用户设置为艺人)**

根据商户ID 查询singer\_base\_info 返回列表



|  |  |
| --- | --- |
| **输入参数** | **描述** |
| merchantid | 商户ID |
| uuid | 聊天室对应的UUID |

|  |  |
| --- | --- |
| **输出参数** | **描述** |
| code | 0 失败 1 成功 |
| message | 失败描述信息 |
| size | 记录行数 |
| data | json的list信息 |

"data": [

{

" id": "1", //记录ID

" openid": "1", //发送消息用户id

" user\_id ": "1", //发送消息用户id

" nickname": "ds", //用户昵称

" user\_picture ": "1.jpg", //用户图像

" sex ": "1", //性别

" type": "1", //艺人类型 1.DJ 2.MC 3.歌手 4.乐队 5.舞蹈 6.主持人 7.灯光师 8.VJ

}

]

## PC投射模块：（TODO）