**超级微信服务端项目接口文档**

目录

[术语翻译 3](#_Toc453166386)

[接口文档说明 3](#_Toc453166387)

[1. 获取服务端状态 3](#_Toc453166388)

[2. 注册用户 4](#_Toc453166389)

[3. 取消注册 5](#_Toc453166390)

[4. 用户登录 5](#_Toc453166391)

[5. 下载用户或群组头像 6](#_Toc453166392)

[6. 更新用户或群组头像 7](#_Toc453166393)

[7. 更新用户昵称 7](#_Toc453166394)

[8. 更新用户密码 8](#_Toc453166395)

[9. 下载好友列表，全部数据 9](#_Toc453166396)

[10. 下载好友列表，分页显示的数据 10](#_Toc453166397)

[11. 添加好友 12](#_Toc453166398)

[12. 删除好友 13](#_Toc453166399)

[13. 根据用户账号查找用户 13](#_Toc453166400)

[14. 根据用户账号或昵称模糊查询用户列表，分页显示的数据 14](#_Toc453166401)

[15. 创建群组 15](#_Toc453166402)

[16. 更新群组名称 16](#_Toc453166403)

[17. 添加群组成员 17](#_Toc453166404)

[18. 添加多个群组成员 18](#_Toc453166405)

[19. 根据群组id下载群组成员列表，全部数据 20](#_Toc453166406)

[20. 根据群组id下载群组成员列表，分页显示的数据 21](#_Toc453166407)

[21. 根据群组环信id下载群组成员列表，全部数据 23](#_Toc453166408)

[22. 根据群组环信id下载群组成员列表，分页显示的数据 24](#_Toc453166409)

[23. 删除群组成员 25](#_Toc453166410)

[24. 删除多个群组成员 26](#_Toc453166411)

[25. 删除群组 27](#_Toc453166412)

[26. 下载指定用户的群组列表，全部数据 28](#_Toc453166413)

[27. 下载全部的公开群组列表，分页显示的数据 29](#_Toc453166414)

[28. 根据群组名称模糊查询群组列表，全部数据 30](#_Toc453166415)

[29. 根据群组id查找群组 31](#_Toc453166416)

[30. 根据群组环信id查找群组 32](#_Toc453166417)

[31. 上传用户的地理位置信息 34](#_Toc453166418)

[32. 更新用户的地理位置信息 34](#_Toc453166419)

[33. 下载附近的用户列表，分页显示的数据 35](#_Toc453166420)

[附录：消息码对应表 37](#_Toc453166421)

# 术语翻译

|  |  |  |
| --- | --- | --- |
| 缩略语 | 中英文全称 | 中文解释 |
| 环信服务器 |  |  |
| 远端服务器 |  |  |
| 环信数据库 |  |  |
| 远端数据库 |  |  |
| 本地数据库 |  |  |

# 接口文档说明

1. 请求参数中，如无特殊说明，均为必填信息

# 获取服务端状态

* 接口功能描述：连接远端服务端，获得服务端状态。
* 方法说明：在首次连接服务端获取数据直接使用，用来检测客户端与服务端的连接状态。
* 提交方式：get
* 接口URL：

url=http://10.0.2.2:8080/SuperWeChatServer/Server?request=server\_status

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | server\_status | 状态请求 |

* 响应的json数据及说明：

{"retCode":0,"retMsg":true,"retData":null}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值说明 |
| 1 | 返回内容编码 | retCode | Integer | 0表示成功  非0表示失败，具体参见消息码对应表表示 |
| 2 | 请求是否成功标志 | retMsg | Boolean | true：表示请求数据成功  false：表示请求数据失败 |
| 3 | 请求到的数据 | retData | Object | null：表示没有请求到数据  非null：则表示请求到的数据 |

注：本项目中的所有的响应，均使用以上3个字段来表示请求是否成功和请求到的内容，下面的接口将不再提供上表，只提供请求成功后的响应的json数据。

# 注册用户

* 接口功能描述：在远端服务器注册新用户。
* 方法说明：将新用户各个字段及头像使用POST提交到远端服务器。
* 提交方式：post
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=register&m\_user\_name=&m\_user\_nick=&m\_user\_password=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | register | 注册用户请求 |
| 2 | 注册用户的帐号 | m\_user\_name | String |  | 只能包含数字字母下划线 |
| 3 | 注册用户的昵称 | m\_user\_nick | String |  | 中文数字字母 |
| 4 | 注册用户的密码 | m\_user\_password | String |  |  |
| 5 | 注册用户的头像 | file | FileBody |  | 使用表单提交的200x200的jpg图片 |

* 响应的json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 取消注册

* 接口功能描述：当客户端注册环信服务器失败后，删除本地注册的用户数据。
* 方法说明：在客户端注册新用户时候，环信服务器注册失败需要将远端提交注册的新用户数据及图片删除。
* 提交方式：get
* 接口URL：

url=http://10.0.2.2:8080/SuperWeChatServer/Server?request=unregister&m\_user\_name=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | unregister | 取消用户注册请求 |
| 2 | 取消注册的用户账号 | m\_user\_name | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 用户登录

* 接口功能描述：用户登录。
* 方法说明：使用用户账号和密码登录远端服务端。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=login&m\_user\_name=&m\_user\_password=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | login | 取消用户注册请求 |
| 2 | 登录的用户账号 | m\_user\_name | String |  |  |
| 3 | 登录的用户密码 | m\_user\_password | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "muserName": "xiaoli",  "muserNick": "lizhenyan",  "mavatarId": 5,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1465366197599"  }  } |

# 下载用户或群组头像

* 接口功能描述：从远端服务器下载用户的头像。
* 方法说明：用户登录成功或者显示其他用户头像时使用此方法。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_avatar& name\_or\_hxid =&avatarType=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_avatar | 下载用户头像的请求 |
| 2 | 下载头像的图片名称 | name\_or\_hxid | String |  | 用户名或群组环信id |
| 3 | 头像类型 | avatarType | String | user\_avatar  或  group\_icon | user\_avatar：用户头像  group\_icon：群组头像 |

* 响应json信息

无

# 更新用户或群组头像

* 接口功能描述：更新用户头像。
* 方法说明：用户提交新头像图片POST的远端服务器。
* 提交方式：post
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=upload\_avatar&avatarType=& name\_or\_hxid=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | upload\_avatar | 更新头像的请求 |
| 2 | 更新头像的类型 | avatarType | String | user\_avatar  或  group\_icon | 上传头像的类型 |
| 3 | 更新头像的图片名称 | name\_or\_hxid | String |  | 用户账号或群组环信ID |
| 4 | 更新的头像 | file | FileBody |  | 使用表单提交的200x200的jpg图片 |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 更新用户昵称

* 接口功能描述：更新用户昵称。
* 方法说明：用户提交新的昵称到远端服务器。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=update\_nick&m\_user\_name=&m\_user\_nick=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | update\_nick | 更新昵称的请求 |
| 2 | 更新昵称的用户账号 | m\_user\_name | String |  |  |
| 3 | 更新的用户昵称 | m\_user\_nick | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "muserName": "xiaoli",  "muserNick": "lilili",  "mavatarId": 5,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1465366197599"  }  } |

# 更新用户密码

* 接口功能描述：更新用户密码。
* 方法说明：用户提交新的密码到远端服务器。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=update\_password&m\_user\_name=&m\_user\_password=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | update\_password | 更新昵称的请求 |
| 2 | 更新昵称的用户账号 | m\_user\_name | String |  |  |
| 3 | 更新的用户密码 | m\_user\_password | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "muserName": "xiaoli",  "muserNick": "lilili",  "mavatarId": 5,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1465366197599"  }  } |

# 下载好友列表，全部数据

* 接口功能描述：下载用户的好友列表的全部数据。
* 方法说明：获取指定账号的好友列表全部数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_contact\_all\_list&m\_contact\_user\_name=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_contact\_all\_list | 下载所有好友数据的请求 |
| 2 | 指定的用户账号 | m\_contact\_user\_name | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": [  {  "muserName": "lisi",  "muserNick": "李思思",  "mavatarId": 2,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455"  },  {  "muserName": "wangwu",  "muserNick": "王五",  "mavatarId": 3,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455"  }  ]  } |

# 下载好友列表，分页显示的数据

* 接口功能描述：下载用户的好友列表的分页数据。
* 方法说明：根据pageid和pagesize获取指定账号的好友列表分页数据。
* 提交方式：get
* 接口URL：

url=http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_contact\_page\_list&m\_contact\_user\_name=&page\_id=&page\_size=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_contact\_page\_list | 下载好友分页数据的请求 |
| 2 | 指定的用户账号 | m\_contact\_user\_name | String |  |  |
| 3 | 分页数据起始下标 | page\_id | Int |  |  |
| 4 | 分页数据每页数据大小 | page\_size | Int |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "currentPage": 1,  "maxRecord": 2,  "pageData": [  {  "muserName": "lisi",  "muserNick": "李思思",  "mavatarId": 2,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455"  },  {  "muserName": "wangwu",  "muserNick": "王五",  "mavatarId": 3,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455"  }  ]  }  } |

# 添加好友

* 接口功能描述：将两个用户添加为好友。
* 方法说明：根据两个用户的用户账号添加为好友关系，增加一条数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=add\_contact&m\_contact\_user\_name=&m\_contact\_cname=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | add\_contact | 添加好友的请求 |
| 2 | 添加好友的用户账号 | m\_contact\_user\_name | String |  |  |
| 3 | 被添加的用户账号 | m\_contact\_cname | String |  |  |

* 响应json数据：（返回的retData为好友信息）

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "muserName": "xiaoli",  "muserNick": "lilili",  "mavatarId": 5,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1465366197599"  }  } |

# 删除好友

* 接口功能描述：将两个用户为好友关系解除。
* 方法说明：根据两个用户的用户账号删除之间好友关系，删除两条数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=delete\_contact&m\_contact\_user\_name=&m\_contact\_cname=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | delete\_contact | 添加好友的请求 |
| 2 | 删除好友的用户账号 | m\_contact\_user\_name | String |  |  |
| 3 | 被删除的用户账号 | m\_contact\_cname | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 根据用户账号查找用户

* 接口功能描述：根据用户账号查找用户数据。
* 方法说明：根据用户账号，查找用户的数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=find\_user&m\_user\_name=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | find\_user | 查找用户的请求 |
| 2 | 查找的用户账号 | m\_user\_name | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "muserName": "zhangsan",  "muserNick": "mynick",  "mavatarId": 1,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455"  }  } |

# 根据用户账号或昵称模糊查询用户列表，分页显示的数据

* 接口功能描述：根据用户账号和昵称查找用户数据，模糊查询。
* 方法说明：根据用户账号和昵称，模糊查找所有符合的用户数据。
* 提交方式：get
* 接口URL：

url=http://10.0.2.2:8080/SuperWeChatServer/Server?request=find\_users\_for\_search&m\_user\_name=&m\_user\_nick=&page\_id=&page\_size=

请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | find\_users\_for\_search | 根据用户名或昵称模糊查找用户 |
| 2 | 查找的用户账号 | m\_user\_name | String |  | 和昵称二选一 |
| 3 | 查找的用户昵称 | m\_user\_nick | String |  | 和账号二选一 |
| 4 | 分页数据起始下标 | page\_id | Int |  |  |
| 5 | 分页数据每页数据大小 | page\_size | Int |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "currentPage": 1,  "maxRecord": 1,  "pageData": [  {  "muserName": "zhangsan",  "muserNick": "mynick",  "mavatarId": 1,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455"  }  ]  }  } |

# 创建群组

* 接口功能描述：在远端服务器创建新的群组。
* 方法说明：将群组各个字段及头像使用POST提交到远端服务器，服务端会创建新的群组，保存群组头像，创建一条群组成员数据。
* 提交方式：post
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=create\_group&m\_group\_hxid=&m\_group\_name=&m\_group\_description=&m\_group\_owner=&m\_group\_is\_public=&m\_group\_allow\_invites=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | create\_group | 注册用户请求 |
| 2 | 环信群组id | m\_group\_hxid | String |  | 注册环信群组后得到的id |
| 3 | 群组名称 | m\_group\_name | String |  | 中文数字字母 |
| 4 | 群组简介 | m\_group\_description | String |  |  |
| 5 | 群组所者人账号 | m\_group\_owner | String |  |  |
| 6 | 群组是否公开 | m\_group\_is\_public | Boolean |  |  |
| 7 | 群组是否开放邀请 | m\_group\_allow\_invites | Boolean |  |  |
| 8 | 新建群组的头像 | file | FileBody |  | 使用表单提交的200x200的jpg图片 |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 更新群组名称

* 接口功能描述：更新群组名称。
* 方法说明：群组提交新的名称到远端服务器。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=update\_group\_name&m\_group\_name=&m\_group\_id=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | update\_group\_name | 更新群组名称的请求 |
| 2 | 新的群组名称 | m\_group\_name | String |  |  |
| 3 | 需要更新的群组id | m\_group\_id | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "mavatarId": 7,  "mavatarUserName": "1234567890",  "mavatarPath": "group\_icon",  "mavatarType": 1,  "mavatarLastUpdateTime": "1465368919301",  "mgroupId": 2,  "mgroupHxid": "1234567890",  "mgroupName": "cloud",  "mgroupDescription": "ucaidesc",  "mgroupOwner": "xiaoli",  "mgroupLastModifiedTime": "1465368919291",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 1,  "mgroupIsPublic": true,  "mgroupAllowInvites": true  }  } |

# 添加群组成员

* 接口功能描述：添加群组成员。
* 方法说明：添加一个群组成员到指定的群组，并更新群组中的成员数量。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=add\_group\_member&m\_member\_user\_name=&m\_member\_group\_hxid=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | add\_group\_member | 添加群组成员的请求 |
| 2 | 添加成员的用户账号 | m\_member\_user\_name | String |  |  |
| 3 | 添加成员的群组环信id | m\_member\_group\_hxid | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "mavatarId": 7,  "mavatarUserName": "1234567890",  "mavatarPath": "group\_icon",  "mavatarType": 1,  "mavatarLastUpdateTime": "1465368919301",  "mgroupId": 2,  "mgroupHxid": "1234567890",  "mgroupName": "cloud",  "mgroupDescription": "ucaidesc",  "mgroupOwner": "xiaoli",  "mgroupLastModifiedTime": "1465368919291",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 2,  "mgroupIsPublic": true,  "mgroupAllowInvites": true  }  } |

# 添加多个群组成员

* 接口功能描述：添加多个群组成员。
* 方法说明：添加多个群组成员到指定的群组，并更新群组中的成员数量。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=add\_group\_members&m\_member\_user\_id=&m\_member\_user\_name=&m\_member\_group\_hxid=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | add\_group\_members | 添加多个群组成员的请求 |
| 2 | 添加成员的用户id集合 | m\_member\_user\_id | String |  | 每个id用逗号分隔连接 |
| 3 | 添加成员的用户账号集合 | m\_member\_user\_name | String |  | 每个账号用逗号分隔连接 |
| 4 | 添加成员的群组环信id | m\_member\_group\_hxid | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "mavatarId": 7,  "mavatarUserName": "1234567890",  "mavatarPath": "group\_icon",  "mavatarType": 1,  "mavatarLastUpdateTime": "1465368919301",  "mgroupId": 2,  "mgroupHxid": "1234567890",  "mgroupName": "cloud",  "mgroupDescription": "ucaidesc",  "mgroupOwner": "xiaoli",  "mgroupLastModifiedTime": "1465369162968",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 4,  "mgroupIsPublic": true,  "mgroupAllowInvites": true  }  } |

# 根据群组id下载群组成员列表，全部数据

* 接口功能描述：下载指定群组的群组成员，全部数据。
* 方法说明：获取指定群组的成员列表全部数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_group\_members\_by\_groupid&m\_member\_group\_id=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_group\_  members\_by\_groupid | 下载所有群组成员数据的请求 |
| 2 | 指定的群组id | m\_member\_group\_id | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": [  {  "muserName": "lisi",  "muserNick": "李思思",  "mavatarId": 2,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mmemberId": 7,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 0  },  ……  {  "muserName": "xiaoli",  "muserNick": "lilili",  "mavatarId": 5,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1465366197599",  "mmemberId": 3,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 1  }  ]  } |

# 根据群组id下载群组成员列表，分页显示的数据

* 接口功能描述：下载指定群组的群组成员，分页数据。
* 方法说明：获取指定群组的成员列表分页数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_group\_members\_by\_limit&m\_member\_group\_id=&page\_id=&page\_size=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_group  \_members\_by\_limit | 下载群组成员分页数据的请求 |
| 2 | 指定的群组id | m\_member\_group\_id | String |  |  |
| 3 | 分页数据起始下标 | page\_id | Int |  |  |
| 4 | 分页数据每页数据大小 | page\_size | Int |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "currentPage": 1,  "maxRecord": 2,  "pageData": [  {  "muserName": "lisi",  "muserNick": "李思思",  "mavatarId": 2,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mmemberId": 7,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 0  },  {  "muserName": "wangwu",  "muserNick": "王五",  "mavatarId": 3,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mmemberId": 8,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 0  }  ]  }  } |

# 根据群组环信id下载群组成员列表，全部数据

* 接口功能描述：根据环信id下载指定群组的群组成员，全部数据。
* 方法说明：根据环信id获取指定群组的成员列表全部数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_group\_members\_by\_hxid&m\_member\_group\_hxid=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_group\_  members\_by\_hxid | 下载所有群组成员数据的请求 |
| 2 | 指定的群组的环信id | m\_member\_group\_hxid | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": [  {  "muserName": "lisi",  "muserNick": "李思思",  "mavatarId": 2,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mmemberId": 7,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 0  },  ……  {  "muserName": "xiaoli",  "muserNick": "lilili",  "mavatarId": 5,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1465366197599",  "mmemberId": 3,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 1  }  ]  } |

# 根据群组环信id下载群组成员列表，分页显示的数据

* 接口功能描述：根据环信id下载指定群组的群组成员，分页数据。
* 方法说明：根据环信id获取指定群组的成员列表分页数据。
* 提交方式：get
* 接口URL：

url=http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_group\_members\_by\_hxid\_limit& m\_member\_group\_hxid=&page\_id=&page\_size=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_group\_  members\_by\_hxid\_limit | 下载群组成员分页数据的请求 |
| 2 | 指定的群组id | m\_member\_  group\_hxid | String |  |  |
| 3 | 分页数据起始下标 | page\_id | Int |  |  |
| 4 | 分页数据每页数据大小 | page\_size | Int |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "currentPage": 1,  "maxRecord": 2,  "pageData": [  {  "muserName": "lisi",  "muserNick": "李思思",  "mavatarId": 2,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mmemberId": 7,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 0  },  {  "muserName": "wangwu",  "muserNick": "王五",  "mavatarId": 3,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mmemberId": 8,  "mmemberGroupId": 2,  "mmemberGroupHxid": "1234567890",  "mmemberPermission": 0  }  ]  }  } |

# 删除群组成员

* 接口功能描述：删除群组成员。
* 方法说明：根据指定的群组和成员信息，删除成员，并更新群组成员数量。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=delete\_group\_member&m\_member\_group\_id=&m\_member\_user\_name=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | delete\_group\_member | 删除群组成员的请求 |
| 2 | 删除成员的群组id | m\_member\_group\_id | String |  |  |
| 3 | 删除成员的用户账号 | m\_member\_user\_name | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "mavatarId": 7,  "mavatarUserName": "1234567890",  "mavatarPath": "group\_icon",  "mavatarType": 1,  "mavatarLastUpdateTime": "1465368919301",  "mgroupId": 2,  "mgroupHxid": "1234567890",  "mgroupName": "cloud",  "mgroupDescription": "ucaidesc",  "mgroupOwner": "xiaoli",  "mgroupLastModifiedTime": "1465369619060",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 3,  "mgroupIsPublic": true,  "mgroupAllowInvites": true  }  } |

# 删除多个群组成员

* 接口功能描述：删除多个群组成员。
* 方法说明：根据指定的群组和成员信息，删除成员，并更新群组成员数量。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=delete\_group\_members&m\_member\_group\_id=&m\_member\_user\_name=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | delete\_group  \_members | 删除群组成员的请求 |
| 2 | 删除成员的群组id | m\_member\_group\_id | String |  |  |
| 3 | 删除成员的用户账号 | m\_member\_user\_name | String |  | 每个账号用逗号分隔连接 |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 删除群组

* 接口功能描述：解散群组。
* 方法说明：删除指定群组的所有群组成员，删除指定的群组数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=delete\_group&m\_group\_id=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | delete\_group | 删除群组的请求 |
| 2 | 删除群组id | m\_group\_id | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 下载指定用户的群组列表，全部数据

* 接口功能描述：下载指定用户的群组列表，全部数据。
* 方法说明：获取指定用户的群组列表全部数据，包含公开群组和非公开群组。
* 提交方式：get
* 接口URL：

url=http://10.0.2.2:8080/SuperWeChatServer/Server?request=find\_group\_by\_user\_name&m\_user\_name=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_groups | 下载所有群组数据的请求 |
| 2 | 指定的用户账号 | m\_user\_name | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": [  {  "mavatarId": 4,  "mavatarUserName": "1460308192724",  "mavatarPath": "group\_avatar",  "mavatarType": 1,  "mavatarLastUpdateTime": "1462870455",  "mgroupId": 1,  "mgroupHxid": "1460308192724",  "mgroupName": "OMG",  "mgroupDescription": "临时测试的群组",  "mgroupOwner": "zhangsan",  "mgroupLastModifiedTime": "1462870455",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 4,  "mgroupIsPublic": true,  "mgroupAllowInvites": false  }  ]  } |

# 下载全部的公开群组列表，分页显示的数据

* 接口功能描述：下载公开群组列表，分页数据。
* 方法说明：获取服务器上面的公开群组列表并分页数据，其中服务器端过滤了用户账号所在的公开群，不在公开群列表显示。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_public\_groups&m\_user\_name=&page\_id=&page\_size=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_  public\_groups | 下载公开群组的分页数据的请求 |
| 2 | 指定的用户账号 | m\_user\_name | String |  |  |
| 3 | 分页数据起始下标 | page\_id | Int |  |  |
| 4 | 分页数据每页数据大小 | page\_size | Int |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "currentPage": 1,  "maxRecord": 1,  "pageData": [  {  "mavatarId": 7,  "mavatarUserName": "1234567890",  "mavatarPath": "group\_icon",  "mavatarType": 1,  "mavatarLastUpdateTime": "1465368919301",  "mgroupId": 3,  "mgroupHxid": "1234567890",  "mgroupName": "ucai",  "mgroupDescription": "ucaidesc",  "mgroupOwner": "xiaoli",  "mgroupLastModifiedTime": "1465371819356",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 1,  "mgroupIsPublic": true,  "mgroupAllowInvites": true  }  ]  }  } |

# 根据群组名称模糊查询群组列表，全部数据

* 接口功能描述：根据群组名称模糊查询群组列表，全部数据。
* 方法说明：根据群组名称模糊查询群组列表全部数据，包含公开群组和非公开群组。
* 提交方式：get
* 接口URL：

url=http://10.0.2.2:8080/SuperWeChatServer/Server?request=find\_group\_by\_group\_name&m\_group\_name=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | find\_group\_by  \_group\_name | 下载所有群组数据的请求 |
| 2 | 指定的用户账号 | m\_group\_name | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": [  {  "mavatarId": 4,  "mavatarUserName": "1460308192724",  "mavatarPath": "group\_avatar",  "mavatarType": 1,  "mavatarLastUpdateTime": "1462870455",  "mgroupId": 1,  "mgroupHxid": "1460308192724",  "mgroupName": "OMG",  "mgroupDescription": "临时测试的群组",  "mgroupOwner": "zhangsan",  "mgroupLastModifiedTime": "1462870455",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 4,  "mgroupIsPublic": true,  "mgroupAllowInvites": false  }  ]  } |

# 根据群组id查找群组

* 接口功能描述：根据群组id查询群组。
* 方法说明：根据群组id查询群组数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=find\_group\_by\_group\_id&m\_group\_id=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | find\_group\_by  \_group\_id | 查询群组数据的请求 |
| 2 | 群组id | m\_group\_id | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": [  {  "mavatarId": 4,  "mavatarUserName": "1460308192724",  "mavatarPath": "group\_avatar",  "mavatarType": 1,  "mavatarLastUpdateTime": "1462870455",  "mgroupId": 1,  "mgroupHxid": "1460308192724",  "mgroupName": "OMG",  "mgroupDescription": "临时测试的群组",  "mgroupOwner": "zhangsan",  "mgroupLastModifiedTime": "1462870455",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 4,  "mgroupIsPublic": true,  "mgroupAllowInvites": false  }  ]  } |

# 根据群组环信id查找群组

* 接口功能描述：根据群组环信id查询群组。
* 方法说明：根据群组环信id查询群组数据。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=find\_group\_by\_group\_hxid&m\_group\_hxid=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | find\_group\_by\_  group\_hxid | 查询群组数据的请求 |
| 2 | 群组环信id | m\_group\_hxid | String |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": [  {  "mavatarId": 4,  "mavatarUserName": "1460308192724",  "mavatarPath": "group\_avatar",  "mavatarType": 1,  "mavatarLastUpdateTime": "1462870455",  "mgroupId": 1,  "mgroupHxid": "1460308192724",  "mgroupName": "OMG",  "mgroupDescription": "临时测试的群组",  "mgroupOwner": "zhangsan",  "mgroupLastModifiedTime": "1462870455",  "mgroupMaxUsers": -1,  "mgroupAffiliationsCount": 4,  "mgroupIsPublic": true,  "mgroupAllowInvites": false  }  ]  } |

# 上传用户的地理位置信息

* 接口功能描述：上传用户地理位置信息。
* 方法说明：提交用户的地理位置数据到远端服务器。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=upload\_location&m\_location\_user\_name=&m\_location\_latitude=&m\_location\_longitude=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | upload\_location | 上传地理位置的请求 |
| 2 | 上传地理位置的用户账号 | m\_location\_user\_name | String |  |  |
| 3 | 上传地理位置的经度 | m\_location\_latitude | Double |  |  |
| 4 | 上传地理位置的纬度 | m\_location\_longitude | Double |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 更新用户的地理位置信息

* 接口功能描述：更新用户地理位置信息。
* 方法说明：更新用户的地理位置数据到远端服务器。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=update\_location&m\_location\_user\_name=&m\_location\_latitude=&m\_location\_longitude=&m\_location\_is\_searched=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | upload\_location | 上传地理位置的请求 |
| 3 | 更新地理位置的用户账号 | m\_location\_user\_name | String |  |  |
| 4 | 更新地理位置的经度 | m\_location\_latitude | Double |  |  |
| 5 | 更新地理位置的纬度 | m\_location\_longitude | Double |  |  |
| 6 | 是否可以被搜索到 | m\_location\_is\_searched | Boolean |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": null  } |

# 下载附近的用户列表，分页显示的数据

* 接口功能描述：根据用户账号查找附近用户的地理位置数据，分页数据。
* 方法说明：根据用户账号，查找附近可以被搜索到的用户地理位置数据集合。
* 提交方式：get
* 接口URL：

url= http://10.0.2.2:8080/SuperWeChatServer/Server?request=download\_location&m\_location\_user\_name=&page\_id=&page\_size=

* 请求参数信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 参数描述 | 参数名称 | 参数类型及长度 | 参数值 | 备注 |
| 1 | 客户端请求类型 | request | String | download\_location | 查找用户的请求 |
| 2 | 指定的用户账号 | m\_location\_  user\_name | String |  |  |
| 3 | 分页数据起始下标 | page\_id | Int |  |  |
| 4 | 分页数据每页数据大小 | page\_size | Int |  |  |

* 响应json数据：

|  |
| --- |
| {  "retCode": 0,  "retMsg": true,  "retData": {  "currentPage": 1,  "maxRecord": 3,  "pageData": [  {  "muserName": "xiaoli",  "muserNick": "lilili",  "mavatarId": 5,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1465366197599",  "mlocationId": 4,  "mlocationLatitude": 39.915215,  "mlocationLongitude": 116.403725,  "mlocationIsSearched": true,  "mlocationLastUpdateTime": "1465372754726",  "distance": 1.79  },  {  "muserName": "wangwu",  "muserNick": "王五",  "mavatarId": 3,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mlocationId": 3,  "mlocationLatitude": 39.915118,  "mlocationLongitude": 116.403962,  "mlocationIsSearched": true,  "mlocationLastUpdateTime": "1465291129532",  "distance": 1.81  },  {  "muserName": "lisi",  "muserNick": "李思思",  "mavatarId": 2,  "mavatarPath": "user\_avatar",  "mavatarType": 0,  "mavatarLastUpdateTime": "1462870455",  "mlocationId": 2,  "mlocationLatitude": 39.932199,  "mlocationLongitude": 116.183566,  "mlocationIsSearched": true,  "mlocationLastUpdateTime": "1465290856356",  "distance": 18.2  }  ]  }  } |

# 附录：消息码对应表

|  |  |  |
| --- | --- | --- |
| 消息码 | 英文全名 | 中文解释 |
| 0 | SUCCESS | 默认成功 |
| 101 | REGISTER\_USERNAME\_EXISTS | 账号已经存在 |
| 102 | REGISTER FAIL | 注册失败 |
| 103 | UNREGISTER FAIL | 解除注册失败 |
| 104 | USER SEARCH FAIL | 查找用户失败 |
| 105 | LOGIN UNKNOW USER | 账户不存在 |
| 106 | LOGIN ERROR PASSWORD | 账户密码错误 |
| 107 | LOGIN SUCCESS | 登陆成功 |
| 108 | USER SAME NICK | 昵称未修改 |
| 109 | USER UPDATE NICK FAIL | 昵称修改失败 |
| 110 | USER SAME PASSWORD | 昵称未修改 |
| 111 | USER UPDATE PASSWORD FAIL | 昵称修改失败 |
| 112 | LOCATION UPLOAD FAIL | 用户上传地理位置失败 |
| 113 | LOCATION UPDATE FAIL | 用户更新地理位置失败 |
| 201 | REGISTER UPLOAD AVATAR FAIL | 上传头像失败 |
| 202 | UPLOAD AVATAR FAIL | 更新头像失败 |
| 301 | CONTACT FIRENDED | 已经是好友关系 |
| 302 | CONTACT ADD FAIL | 好友关系添加失败 |
| 303 | CONTACT DEL FAIL | 好友关系删除失败 |
| 304 | GET CONTACT ALL FAIL | 获取全部好友列表失败 |
| 305 | GET CONTACT PAGES FAIL | 分页获取好友列表失败 |
| 401 | GROUP HXID EXISTS | 群组环信ID已经存在 |
| 402 | GROUP CREATE FAIL | 创建群组失败 |
| 403 | GROUP ADD MEMBER FAIL | 添加群组成员失败 |
| 404 | GROUP GET MEMBERS FAIL | 获取群成员失败 |
| 405 | GROUP UNKONW | 群组不存在 |
| 406 | GROUP SAME NAME | 有相同群组名称 |
| 407 | GROUP UPDATE NAME FAIL | 群组名称修改失败 |
| 408 | GROUP DELETE MEMBER FAIL | 删除群组成员失败 |
| 409 | GROUP DELETE MEMBERS FAIL | 删除多群组成员失败 |
| 410 | GROUP DELETE FAIL | 删除群组失败 |
| 411 | GROUP FIND BY GOURP ID FAIL | 根据群组id查找群组失败 |
| 412 | GROUP FIND BY HX ID FAIL | 根据环信id查找群组失败 |
| 413 | GROUP FIND BY USER NAME FAIL | 查找用户名称查找查找群组失败 |
| 414 | GROUP FIND BY GROUP NAME FAIL | 查找群组名称查找查找群组失败 |
| 415 | PUBLIC GROUP FAIL | 查找公开群失败 |
| 501 | LOCATION GET FAIL | 获取附近的人失败 |
| 999 | UNKNOW | 未知错误 |
| -1 | ILLEGAL REQUEST | 非法请求 |