

FlyBook即时通信服务端接口文档

更新日期：2025.12.2

0 准备工作

在开始之前，请确保项目已经启动（默认端口 8081）。

首先需要建SQL表。创建数据库flybook，在application.yaml中配置密码、数据库名。创建表：

代码块

```
1  create table conversation_members
2  (
3      id          bigint auto_increment
4          primary key,
5      conversation_id bigint
6          user_id      bigint
7      last_ack_seq  bigint    default 0
        到的序列号',
8      unread_count   int      default 0
        未读消息数',
9      role          tinyint   default 1
        成员, 2=管理员',
10     is_muted       tinyint   default 0
        免打扰',
11     is_pinned      tinyint   default 0
        是否置顶',
12     joined_time    timestamp default CURRENT_TIMESTAMP null,
13     constraint uk_conv_user
14         unique (conversation_id, user_id)
15 )
16     comment '会话成员表';
17
18 create table conversations
19 (
20     conversation_id bigint auto_increment
21         primary key,
22     type          tinyint   default 1
        not null comment
        '类型: 1=单聊(P2P), 2=群聊(Group)',
23     name          varchar(128)
        群名',
24     avatar_url    varchar(255)
        群头像',
25     owner_id      bigint    default 0
        群主ID',
```

```
26      current_seq      bigint      default 0          null comment '当前会
27     话最新序列号',
28      last_msg_content varchar(512) default ''        null comment '最新消
29     息摘要',
30      last_msg_time    timestamp   default CURRENT_TIMESTAMP null comment '最新消
31     息时间',
32      created_time     timestamp   default CURRENT_TIMESTAMP null
33  )
34      comment '会话表';
35
36
37  create table message_reactions
38  (
39      id          bigint auto_increment
40          primary key,
41      message_id  bigint
42          not null comment '关联消息
43      ID',
44      user_id     bigint
45          not null comment '操作用户
46      ID',
47      reaction_type varchar(32)
48          not null comment '表情代
49      码, 如: thumbsup, heart',
50      created_time timestamp default CURRENT_TIMESTAMP null,
51      constraint uk_msg_user_react
52          unique (message_id, user_id, reaction_type)
53  )
54      comment '消息表情回应表';
55
56
57  create table messages
58  (
59      message_id    bigint auto_increment comment '全局唯一消息ID'
60          primary key,
61      conversation_id bigint
62          not null,
63      sender_id     bigint
64          not null,
65      seq           bigint
66          not null comment '会话内
67      序列号',
68      quote_id      bigint   default 0
69          null comment '引用的消息
70      ID (若不为0, 则是回复消息)',
71      msg_type      tinyint
72          not null comment
73      '1=Text, 2=Image, 3=Video, 4=File, 5=TodoCard, 6=RichPost',
74      content       json
75          not null comment '消息内
76      容载体',
77      mentions      json
78          null comment '被@用户ID
79      列表',
80      is_revoked    tinyint  default 0
81          null comment '是否已撤
82      回',
83      created_time  timestamp default CURRENT_TIMESTAMP null,
84      constraint uk_conv_seq
85          unique (conversation_id, seq)
```

```
61  )
62      comment '消息存储表';
63
64  create index idx_conv_time
65      on messages (conversation_id, created_time);
66
67  create table users
68  (
69      user_id      bigint auto_increment comment '用户唯一ID'
70          primary key,
71      username     varchar(64)                      not null comment '用户名/姓名',
72      avatar_url   varchar(255) default ''           null comment '头像链接',
73      password     varchar(255)                      null comment '密码 (可为空, 用于测试)',
74      created_time timestamp    default CURRENT_TIMESTAMP null
75  )
76      comment '用户信息表';
```

在数据库执行以下 SQL，以模拟用户：

代码块

```
1  -- 1. 清理旧数据 (防止 ID 冲突或数据干扰)
2  DELETE FROM messages;
3  DELETE FROM conversation_members;
4  DELETE FROM conversations;
5  DELETE FROM users;
6
7  -- 2. 创建三个测试用户
8  -- 用户 A: ZhangSan (ID: 1001) -> 扮演群主/发送者
9  -- 用户 B: LiSi      (ID: 1002) -> 扮演接收者/被邀请者
10 -- 用户 C: WangWu    (ID: 1003) -> 扮演路人
11
12 -- 注意: password 字段设置为 NULL, 表示测试用户没有密码, 登录时传空字符串即可
13 INSERT INTO users (user_id, username, avatar_url, password) VALUES
14 (1001, 'ZhangSan', 'https://api.dicebear.com/7.x/avatars/svg?seed=Zhang', NULL),
15 (1002, 'LiSi', 'https://api.dicebear.com/7.x/avatars/svg?seed=Li', NULL),
16 (1003, 'WangWu', 'https://api.dicebear.com/7.x/avatars/svg?seed=Wang', NULL);
```

🎬 场景脚本：一次完整的群聊体验

我们将模拟以下剧情：

1. LiSi (1002) 打开手机（连接 WebSocket），准备接收消息。
 2. ZhangSan (1001) 创建了一个叫“周末去哪玩”的群。
 3. ZhangSan 把 LiSi 拉进群。
 4. ZhangSan 在群里发了一句“Hello”。
 5. LiSi 收到推送，并刷新列表看到未读消息。
-

1 用户上线 (WebSocket)

这里也可以多开几个ws窗口 多建立几个在线用户，发信息选的是http，服务端推送选的是ws协议。

我们需要一个工具来模拟手机端的长连接，看着服务器推什么数据过来。

- 工具：可使用在线工具 [PieSocket](#) 或 [WebSocket King](#)（或者直接用 Apifox 的 WebSocket 功能）。
- 操作：
 - a. 在浏览器打开上述工具。
 - b. 输入地址：`ws://localhost:8081/ws/1002`，其中1002是LiSi的user_id。
 - c. 点击 Connect。
- 预期：
 - 工具显示状态为 `Connected`（绿色）。
 - 后端控制台日志输出：

代码块

```
1  用户 connected: 1002, 当前在线人数: 1
2  用户 connected: 1001, 当前在线人数: 2
```

⚠ 注意：测试过程中请不要关闭这个网页窗口，把它挂在旁边。

2 登录 (HTTP) (可选)

如果在application.yaml中启用了登录，则需登录：

代码块

```
1  # 认证配置
2  auth:
3    enabled: false
```

- 接口: POST <http://localhost:8081/api/users/login>
- Headers: `Authorization` Bearer 1001
- Body:

代码块

```
1  {
2      "username": "ZhangSan",
3      "password": ""
4 }
```

返回示例:

代码块

```
1  {
2      "code": 0,
3      "msg": "success",
4      "data": {
5          "userInfo": {
6              "userId": 1001,
7              "username": "ZhangSan",
8              "avatarUrl": "https://api.dicebear.com/7.x/avatars/svg?seed=Zhang",
9              "password": null,
10             "createdTime": "2025-11-29T17:03:34"
11         },
12         "token":
13             "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiIxMDAxIiwiaWF0IjoxNzY0NDExMzAxLCJleHAiOjE3NjUwMTYxMDF9.TY4dhmnW5Q3AhlofGMVkhG0xliraofv9Ucnh7Unzm08"
14     }
}
```

3 创建群聊 (HTTP)

现在切换到 Apifox，模拟 ZhangSan (1001) 创建群组。

- 接口: POST <http://localhost:8081/api/conversations/create>
- Headers:
 - 无登录: X-User-Id: 1001

- 登录: Headers: `Authorization` Bearer
eyJhbGci...iJ9.eyJzd...MDF9.TY4d...m08

3.1 指定成员

在创建群聊的同时指定目标成员。

- Body (JSON):

代码块

```
1  {
2      "type": 2, // 普通群聊, 预留其他类型
3      "name": "周末去哪玩"
4      "targetUserIds": [1003] // 目标成员列表
5  }
```

发送请求后, 服务端将查询是否已有同名、同成员的群, 如果有, 则返回已有的群的ID, 否则返回新群的ID:

代码块

```
1  {
2      "code": 0,
3      "msg": "success",
4      "data": 2
5  }
```

其中 `data` (即 `conversationId`)是群聊ID。

3.2 不指定成员

创建一个空群, 稍后调用接口拉人。

- Body (JSON):

代码块

```
1  {
2      "type": 2,
3      "name": "周末去哪玩"
4  }
```

- 预期:

```
1  {
2      "code": 0,
3      "msg": "success",
4      "data": 1
5 }
```

4 拉人入群 (HTTP)

ZhangSan (1001) 将 LiSi (1002) 拉入群聊。这是检验“系统通知”的关键步骤。

- 接口: POST <http://localhost:8081/api/conversations/members/add>
- Headers:
 - 无登录: X-User-Id: 1001
 - 登录: Headers: `Authorization` Bearer eyJhbGci...iJ9eyJzd...MDF9.TY4d...m08
- Body (JSON):

代码块

```
1  {
2      "conversationId": 1,
3      "targetUserIds": [1002]
4 }
5
```

- 预期:

代码块

```
1  {
2      "code": 0,
3      "msg": "success",
4      "data": null
5 }
```

在ZhangSan (1001) 的WebSocket端收到 JSON 推送:

代码块

```
1  {
2      "msgType": 1,
```

```
3     "conversationId": 1,
4     "messageId": 1,
5     "content": "{\"text\": \"ZhangSan 邀请 Lisi 加入了群聊\"}",
6     "senderId": 1001,
7     "createdTime": 1764420831006,
8     "seq": 1
9 }
```

5 发送消息 (HTTP)

5.1 发送文本

ZhangSan (1001) 发言。

- 接口: POST <http://localhost:8081/api/messages/send>
- Headers:
 - 无登录: X-User-Id: 1001
 - 登录: Headers: `Authorization` Bearer eyJhbGci...iJ9eyJd...MDF9.TY4d...m08
- Body (JSON):

代码块

```
1 {
2     "conversationId": 1,
3     "text": "Hello"
4 }
```

- 预期:

a. Apifox 返回成功，且 seq 增加：

代码块

```
1 {
2     "code": 0,
3     "msg": "success",
4     "data": {
5         "messageId": 3,
6         "conversationId": 1,
7         "SenderId": 1001,
8         "seq": 2,
```

```

9      "quoteId": null,
10     "msgType": 1,
11     "content": "{\"text\":\"Hello\"}",
12     "mentions": null,
13     "isRevoked": null,
14     "createdTime": "2025-11-29T21:10:19.0016336"
15   }
16 }
```

- b. 查看 LiSi 的 WebSocket 网页，应该收到：

代码块

```

1  {
2    "msgType": 1,
3    "conversationId": 1,
4    "messageId": 3,
5    "content": "{\"text\":\"Hello\"}",
6    "senderId": 1001,
7    "createdTime": 1764421810379,
8    "seq": 2
9  }
```

5.2 发送图片/视频/TODO

- 只有Body的不同，在msgType定义消息类型；在content定义内容。目前在content中传递JSON字符串，与服务端解耦，客户端可以根据自己的需要，自由地解析其内容。
- 如图片采用base64编码：

代码块

```

1  {
2    "conversationId": 1,
3    "msgType": 2 // 图片
4    "content": "{\"base64\":
\"...\"}
```

akkADC9/M85+Fh3nnfmWyyZ2fnLUMEAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AA
AA
AAAAAAAAAAAAAAAAAAZzxFv+FkLRLp8bZIV6u+oOU5SFV/1W88K4i0IpEL9ZtrhRFRWylQsKrjiGh2zb95q
qrm6vcFOGMsZH4hRkToFEawiCyw2z835uWdrz+fswLzaP3P7BmqiISKSLSI9BCRYSIygLmwN/PHi5m3
3M48JZM5Ud8swGm1gHlFax9/0C0Zd5aXc0lq6qGCrKxJ+vM5KzCP1P/MDQn1eCKNEZG+zNaJzB//mXk
OM3fQHwLglFsosvxUJOSZNpsrPzU1zZyWNrv0iYiEEfMYxHGPIZEwfZ/0zWxGShjyARaS8iI5kz7x
N5Eknj9HqJuZN+2zn1vCaLUQ6HMzUj45g1Pf0e/XkQ83n6/WsjREQmVVUt46NHo/TVzkjNlCw8I1pER
jNnPm08R0RU/SGh5Z3P/NBo5nVy6FCIvxuM9If9I0lkpr8VFVk6Mf+nVXz8y/pyIhL9hlouIppgty8i
RRmgLztXWIhol6L0Xq8ob81nfltEgvX7QMtkJararyg3z+7b9/Ny5gR9+ZnmD5ks0orQiwUF1r4u19b
c60iliqK49fsQEZEI6zfVEiI3MfstP1fkEtE7qnrtUJF3D4m00pdDi9IqiegTVb10rsj/vnU4+ul30J
P84ZJF0BE9YzSwj6uq+p+pdes/9+/Uyabfp4H8XnmcaxxEdFBRps0VeWcvrjb00QcRbVXV7ncZDJ8/7
HSesVe7zT70YqHI8s1ED1XoC4iok6a5Hzazcs0ZQ7UmHrsDc9DYqqpMNSztk7du+/Xl59Qc1mXQori
MyGuMpsLko3GWwq/frv0ZWcc5pGkKLv1m4mIQono7tLS4v0cztDykJDWJo0B0g0Gyo2MVHKIqFhfIYB
QIrpk097fkps7R4mLc+jLoZkxLyZF+Yfnpp4iJclED9ytqus9t58JmvSGDSRQskjUNP4+K+vDdnb7Cn
1Zg4kYsqKjS+lj41P1RXwci8kiio2FBYevNxofJQuP9RCQ4PIY0iUGxo66Mfg4D57w8JG7GzduucvR
GTXV/ahGxEtrqy88+HISF/3gaA5+UgWRET9iWgm8/KnVPURRVH03ivgQL0hicxyNDt7g750i2BOIGZN
fw61scpkycjR0brq52RAvSGRInIu0bjfv7lf5/doPzrr10LcnNnLS0v3zLG7ZYgh23oY7zLVcwFBT3
1bUEzY16sf+1r4yIROWIyjdJX0Z18fuq2qJru/ma/ojmXiYihihIdH6rcTEamDB+d36d793cdiYmavNh
rHTXI6d9TXX3w40LjjW6Gh9+m3w6mjERE5HPP120+nU58s4LQZ2aXLtx8XFk6+WtM+CdcXeigmoiMGw
/W8b19HfRmcGgoRqSJn1PgLJItzjBIX59gQFDR7oMhhfZmnLKcgDhQRMu/i/HU6pM6prH8niHKQoiqML
0YpQfYGHYkVRHKo6WL8dzl1IFueoXUSHA13jaqqqVDF30W+HcxeSxTlqEtGCSv1GD9HMQiIG/XY4dyF
ZnI00Mffbt3STpi/w0E/TlIigoHL9djh3IVmcY0Sk9XSilfsUxWdXKx2/Ez/EZisPVdUD+jI4dyFZnE
M2Ml81XmT3EUWZEuiqYoCm0RiHo8BFtE9fBucuJIs/ODl+JSAul8+ZtyIS8pzIxAnMG5cQvb9DUfq49
DvpzCstdcVWV3+7Ky4uQ18G565mTxarIk5/wzNVImrvdlef6ePds5j7WltwFaP/E+kTypxE/m0IrkkQ
7v8naEQUzswUHBzBzEM0Mi9zJw0TW2rcyf9hHZ82+iL/6nKDcWKEq9NyyTqqr42oqKTEdIyJtJ/qb
+NxNm7vQuC9J45qR2zEmhzEk9mZNuZE76njlJRPz+3C2Nmbu9xZx0CXNS6+PnFs+cdMvxc9Pvfzq9zh
z/9cxfxSh85SQfMzn5q8eZT3pErr/3tV8PMvd9iaiXq2ZmrZeLiRb+oig3+Jq01F2E7jGZ/vdQ+/bPa
CcxRTxWUTIKFSVhv720Jk4kE5GgPxHt0ZgtbwYfjfcsaw5m5u7TiA7tUpRofyNu7nM4fnSyP39a64SE
NCIiYo6eYrNdP8NqfVpq3sBCRBSpadYDkZFtt0VGRpYaDDmdFeV8RVEMRiIqJKLqRo7q6a1p8lZ+vrl
/dfXfoxMSntaXNwcR6fwo0bTfRK7dR5QQoSg984nIdjzxhRBRzPGVys1EP48Rye1Ft04ZrfLUUZRA35
50mpm5+/NEMw8RXXWEqG+YonSpPTc+Piu3DRF1JqIykd0XE2V3I1r3rKp+rQfxMJCMiGk1E6/LyX
u/Tvfuf9GWe3mB+4CVFWeXv5hIT0QQiWmk0ru3HvCw8NvaYfp8W83fmYZM0zRTsY/LLqYrpXPkhf+t7
4tQJ9U0kqzDl8wHviWT/ZZ7QS0Rai8jjbLlvX36yrmX+R6iP86mNEU4HZ2VkJh9LD7MBw4MkwMHhvL
+/UM4a1fffC5+aCy7XPo2mhI93W5ta16e0ZKauv5oSUl900cajZk73c28eiBzdtvj_iwnrz8FXqCLStw
bF8r33MN/SElcb0cxdb2f+z3/mgtaNPLduIjkcedcS5jn6dusVYCLZaBH5LTd3rb6Kp/8w3znMR13PG
0JwuvZnZeWUpqauPHqSy0g2+sqCiCivouL8J5WIrVtSDQkVZbovMw25XmjSUX/3RCcrPRGRfb3a+8L/
6Mt02MEJ5A5wZdHRXJAczZylxv1+ZSEiQTft1c83VSmXairRUHY7nnfbL0y6Mia9bu2mee0nTlpqoe3
Z4vvrQIiBaH3vQstVLttrEfHnPRDo61pBAQ+/7ih9/J2idGrU5YMPw+KrtTUdC/MH2+zB7BFtHo+Kij
Xq92kqEYl49i2Z99/W9PBvnZWerurjq734/cl8UIioFVE7IhqSL7/8XyU9MHeuuk0/W20JSMyj6+XmT
9vRktROSpdKJxE5m3BuUUTtmWhkvuy5men02bPVvfrdfPqSFxP5ubJoTbSuQ97rfXr5vrL47ze84EkX
vbo/SFF8/v6dRAN6u2ltdF5hYqV7szkh4c9xiuLz20CL4/fs3W64XFsfRSJ0imPpY5Xcf0Ba/TnVxQl
Er0nr1saqZ035zHWvLD75RCYmnv f7PiqJzJxZvUsku43nfk3BzKETL+M9qo9zqY0502xszkbvY6zIPU
9f350IhE2dyimKjzaaEnHtq90P32uzbdjoWNKcn9wiEnHHHfxZr17ex2xqKKpIXBxbVq/ml09moeGsl
I1ZtIh/6NrV+xhNjSCDSGIim195hVfpj+cbL9a3URujB4r89pvvK4vPP+dbL77Yu45n90nG8sNn2Xzr
cdprqsZTtL6N02LG1dXPdon1PtMwjKeftovI4bpL/3upJ1msqpsssRCToppt4R1BQ3f06dRJJTq56rm7

bjbd6Na9o1877PE4cp6PG03/INTnt6bfr63oSkeDLL+ctBoN3GycbnTqJJCXxgRUr+BYRCdMfuzF27JBWo0bxZ2Fh3sdpjoile3n0Ud4g8kSjE8Y//ylthg/nH0JCvNttjkhMFHniiep/6o/rLUCyGOU7WwzdypcNH84u/f6e0bMHy5df5Nit1vQ3c3LK2+rb0K2WLHE+26eP90m3VLREsvjoI57k72dITHRXv/tu2fS67TdcTg53HTKEK/TtesZjj5U6bLbU/5aUsN/vlcwc+sADvKVtW+/6zRldu4rMmsU/b9/0E/Tn0BA50Rw+eDBvDw72brs5IzzWZNmy6g2NWZl83TqJGTCAd7ZEsvWMnj1FVq1y/jPw1U+AZDHa01ls2cKDBwzgIV2+ntGpk8iHH+bZrdaUzenphWfmEgNr1lQtHTbM++RbIloiwVxzDb+rqt771cb48dVpzHnt6h6jfISP G8ebwr0KTZ0aLWkp2ccs1qPBXxzrl7N/+je3bt+S8UFF7Dj4Yd5cW0uMpg5auZM3hYa6t1eS0Tv3iJP P+28RX8evjBz20mTwu6KQh99+oisWOEI8CHT8GSxfLn06NeP0/X7eUabNiJvv11gs1pTPkpJyela91h nmPXrK2f36KF1BXrTNUc8+aSj2ZPFihXyZFyc9361ERYmkpxs3173GPV74w2e5u+KhUgkKEhk/fpCa1 VV2kuBVhBbuJAndenC1fr6wcEiF15YVXXRRTb3mDF2bezYSpk4UWT8eJFRo0T69RNp3dr7uA2N+HiR5 cv5I5GsBiWMRx7hVe3be7fjKy68kGX+fKusWGFyr11b6HrjjTzLa68VuZYtK3XeeKNNBg+ueX309fTR v79W8e0P5jh6c9G75x5+PTrau76vGDRIk4ULLfL88yXutWsLnW++mWf597+LXE89Veq87jq7DBggij e9fQxbFh1cU50kZ+HWgdOFke05L50RLRsGcf27cuHA72vIiNF1q4ttlItKZ9mZ2f31h+p0fj942yqBQ v46c2blccsFn0JUef0mvuJJ4zZ4eEcrGlKk4+dmMitevSovKNhj8R39GW/0z70gvz0hqwyFyQnG2epa k1viIgYZs3iVz75RP2T06nfu0aXLsL332+79YEhol/Tl/nCzKFJSFTjN9/4f5jRNddU8po1BftDQiLn REd30aoVJyL6y1+473vv0d6jR5UIfVnHjkTvv59/aPjwim8rK4M0IkWtrlZauVyqwl4eRGVlocEFBwp 0fr6h/4EDod00H26lHj1CZPc1EMaPzp2J7r1X+/jBBzNnq+r5f14doo8+4iEPPkg//PabEjCx90LTzY sXV1Refrm1qEsX187Iy0AfFYX2E1lsRMEkIkIU2vHIkchLNm6MGL91a8yYA/4GEhCRwUA0b57jp9dfz xirKP19D1Bdu5bHPfMMFZmZ6bsnqtagQdXaHXeUVk2YUJnXsWP1zsjIoB+j5KCi2Cp/P7fwLvv2RYzb tCl8wtatMc00+vvt1QgPJ5o/v/KLl19eeaWiPKXrtwjQGzKaaMuWY3/Ny0i54q676LMjR5QBbj9D5IK DiVatMlXMn1+6r7IybHFsbM+ACxudMW69VZb7+yRLTNQ0ozHjPZFFB51M200Hh//yS2rgcRaNvLKgmjd3u6uvZn0gDH7RRS7bwYPm7p71/HnySV4RE+PdRm106CDy7be5ZRUvV92vr+vJY0Ax+rq1ERsrsm1b4 Q+BviqIHApHptyT+dCI1NT8B19+2bx91iyHtTFfaXr0EHngAffN+rY9XXYZ/6e+T9sZM+xy8GBWgcOR stblyhihb80X774zXnvxxa7DgW6WxseLfPCBCaG+bq1x43h7fec2d65VUlIycx20lH/abFn1rhIm8oRh2zbTzSNHammBvtr07y/y5ZfFs/T1A11ZjBnj1jIy8t4fM4YPBLq/oqoiy5aZrRZLyq7i4rx6r670KA sXBkoWLEe0ZL2lr9MyGp8sq0Zrw4j4eLbq96+NoCCRG2907BIpDvj0rv370fGCCwlf1HzooTKn1Zrya VLZVoy+f108Ul+3NmJjRT79tHBnY7t3mVP6btxY8ey0aa68QAnNMwYM00zffmsaqW+Li0ivf+XPBXuy Q1/HM665xi55ee15FRVpyVLZ/p0bLyUlJVHDh2tfBURkixbz0ph3e3UTLl3Kl3Ttym79/p5x881WKS5 OyyovT71t796G3zAlIvrii4p2gwZpe/wlo6Agkvus/zK/Ffd4mT+k0X//i7t6qurONDPSyTyl7+US1 lZ6sHi4uzL67b9B1Bfsjh69NimQN/Nm0/TkgUR0Ysv8l0xAbqDY2NFHn/c9oS+Xi0RCZ47lz809GkzY EC1pKRk5lRUpE/R1/dWf7JgzmlSFxlzSteHhrK/1b0nd9v6UFWRefPs34rs8Hqj3347bwp0f6Ffv2rt yJHMfKs1/V593YbKz0TEES04Ut92bYwaJbJvX+5N+no33cTbAp3b80E0LSSrM7usLG2Bvm5Dff89Dxs 40H+XZlKSSHZ23lV1a/lPFv4Sj2fce2+5mEypKwbszQb8DZ2BzoZkISLq90n8dqA7+oMGae7PPiu/TF +XapLN1YFulqqyOuvF7oqK9P+tXlzQwZBtVyyqHXnnC7r0ndmi759fVx4oci0HSVzPesyc9uhQzlNv 29tBAWJrFlTbHM4U18h0rnF/V138f3+uo4jIkRef934hefAMmbu2L8/F+j3rY2QEJFNmwosTmfq83WP 1HgLFmjL/N1AbdNGZPPmonfr/u37Txb1xYIFFWw0pquebzZn130g/g50NyYJqEkaviRPZpK/nGVdcUf2 r/k0qIjETJnCWfl/PmDq10m00pu+zWhP83CXXa/lkQUSUn0xeVN+IRlUVWbq0bLfi3h0X6l9/zZMHDf LetzYGDnRpWVmZB63WnP51j9h4zBwVF+c/MSUnV5SKH0pRu/8HH/CcQL1RF13kcBcWZvxUVXUsru6RG q+ggDt07+4/MT32WHku8xce3e9NSxbXX2/VCgrSM0tLMxo/H+Uk+OwpACJFUY5Nnkuzzj9fX/K7XbsM gx56qMNfPbfdd5888tNPSi/PbZ7aty6915TzUQEbYyK6nFQX346Pf980PqLL5av9Ns9MRMdPBg8VCR yY0225csp6FiAuYyXXmqn2Njq76KiehzSlzWwqqrWyy8nvw/DzsgIiRQJG1r7/xdeIEN+ft19PE2YYJ eOHd1fhYf3ytKXNVaXLmrJuHFUpd9eKzPT0EZReniumN7oD80pUyq1FSuKc0ND5dm2beM36stbEpJFA Pffr343b56sjvFz+9FiIXrnndAb3nijbAER0Z49PHrrVrrNV7dxrZtuKq8eMcLxY2lpindQ3tNNURTh rFn04nkBZ6YQFRZGGAoKwk98Ty4vp+LISPE1nYmIiAY0dFaFhp78xK9aRUX0psFPB6jTGRziNkrDa// PTAwt/NyKvhSiAQMcVkrVrv9aNXNVBAW30d27V1ZGRdnwviWSRkEBBoYGe6Bz6aW06hdFLCps25Zeat fuvAZ13zcnJIt6PPaYev/ll8vXYV639Grk5ipBr7/eamVZWVXcSy/R09nZit/h2n36aLRgQVkxUfDLP Xr0KNWXnwmuuUb5VlHkf/12TzabQoWFakLt/3fvVn+88kptdp8+7DWrMSyMqF07d7GI9n6/ouLKSPK

z6tcVUVUUaGc6BH54Qf1i6QkuSkhQfNKZq1bE7VtqxUQWQKMlGic8nIlM8JrNEyNykois/n3c3vwQXp
v6FD5KSTQMxm0GznSKa+9VpjbpQuvff75+l/py08FJIsGePVVenDIEC7Rb6/13XeGNosXKz9u365MdP
kcElTzKxbPPaXVxQ5t/38c6+P9eVnCkVRbJGRVBJoyJzTSeR2G1p53ntauzb4/eRkx7WXxqo5P0tGR
RGFhytWoLD71tPTmwshQb5uS3sdh0JqhWezrhpk7rh9tur5o0ZUzdhtG5NFBamVChKZC0GqAXWub0E
qn7eVW43kaL8PrBu0SI1789/tk3u10/70biezlqHI0jJyQnLjoo678mnnlK8Et+p40fHAK8xMepPU6Y
oc/r00ZfUECEqKjK0Hzf07+BGuuyyKm3GjIoUpzPylaSkRi9XF+Ct2/xsNvK78jcdHzEYFuY01S/3t3
Bh5Mf/+pfz2iltJLP2j1+EyOVSrIqi+f0u31gHDpDL4XUNUyM0lCgszPt5J8nJUZtWrqy6afx4zqtNN
CJEemmYoJ3L7aa3xUlPJ5e8DIZycKCys7hi0ad0iTuuWBF153nm8118CJCLav99Ac+d2HrxihQSYa9Ky
kCwa6JFH1K+mTZNX/H3/jYlx0dasMa2f0JGL9J/KMTFEycml9lat+J02bboGvMT3ZeFCkoQTF/0t6/v
vZZCi0DAJsABMZCRTu3buqtqHfxVq2zfy402b1JFjxmrgfRUYSmc01yUfE0Wx/a5dcQhNtfm5xtmrllr
ZtXT7vGo0dG7VxxYqqEaNGaTvCwoiKioisVq2VSFCzJeNBg2h8pZ+nN7Vq5ZLWrau9zm30HMV0552VU
xITzb/+b8dTQYES88or8s6zz/JUfdmp0Gy/wL0doi18cqV6+8SJ/JPvS0ZDWLt2Vd//+9/qpMmTudTz
lz5njtu5apT9V5HWTboptWYNHbr3XlnfvUGDzE/0unVyX3Z24Lkd3bo5KCamutDfSl7R0Yrp66/3zZo
zx/Vh+/ZE6ekhvRXFELDNxjCb6VL9tlqd01dyelhm1m+vNXx4q8Lvvv+eev311V9FRRGlp4fEE1kacN
egYSwWuli/rVanTnZ3cLDb572q0+6IMq5cqwcPFgC9CsRZWUpoWvX0jsPP1x9yhcNRrJop0XLlcX9+
rFJv51IyGYztIuLU/b9/e+0yV0mcCkRUe/emtx6q8mkqsH/iorqXKyv1RCqqtoXL1Zunj2b17dr9CT5
hlu7lmdt307X+ptIV2vEiEpHWJgasKtRUYZVv/pq6PQ77+S1Vm4uIjaLM9NTUvj0ZmZ5HdW5ZAhDs1
gCA74h10Urpxr1j1z+cKF7s1GY0g7oshmWcl9zx6+Ij0TfLa1KERDhrhcRKHZ+rJaU6cqmcuXu28YPV
q8rj48ZWQokW+/rb67YIHrRBfxH9LZMigrkNWr+YbzPJbfIxKZ0dMpFkvmQ7X7pKbaR02erJleftkmd
nvqusbON/BFRJQpU6o3hIf/ftzmGpT15Zc8a9w4/80Va6NzZ5Fdu7JzmNPqvHYPPcQ3tWrFe4j4h99D
vg405h1Dhs7t2wxL/Xcv6kWL0DnIyK8z6v23HbuPJbDnHJiDAgR0eLFvDgsjHfXPTfeERrK340ZU82
ff24M0JGvoW68kdf6G/Xbu7fI/v1ZKcxp9X6h/0EHmT5uHHu1oY9evbSi+5y/nGfdH8uJAsRURcs4B
c9Z0HqkwURUVlZVVxm5rFl+fnZYz23nwwRUcaN409q5zqcbLIQkZAnnuA7hgzxXivDV9x2m0WzWFI/L
NGtAP7JJzxywADv/Wvj8sudJubsk7q6+0YbHjNsmp/zv0GGKikrS/1UP8lvwwaeEGgU51VXuXJFMlp7
1mms99/nSRdeyKxvuzZuvdUmVmvaZubUBo2s+04716IJ7zb8QxFEUlMdBc/99wfNGEEShZxcSKHD//
xkwUR0fz5vLS+ZNFSmLnVjtDk0R1fRq2pyeKFF3jEtGm8tUcP79fHV3TvlJnzzGz1Zp+vb4tIqIbbp
A3/E2nNhhELr1U+0gk388ohMCYue3o0bzT3wSr4GCRLVsK7C5X2l36ukRE06fzh/4mk4WGikyYoG2SJ
j41/sgRjh02jPfr262NyEiRbdtyLU5nesAp/nrbt7tvq29hXkURGTTIXbR5s7PeKfVnnEDJYtq0asnP
z9RNpmkpLZcsPv2UxyUksNOzvVOZLIIIMjNd48a0lfL27WvXs6hvmnsNZr7wxRd58XXX8c7YWWh7e/P
pQ1FEVq820Wy2tPeKi31Pz9+zhy8YPJht+rq1ERYmMmGctm/TJvY5Ac+fH3+U3hMn8g+eX7/0cdVVVV
xUlPpzVvWhz6H2n3zCI/v2rfs784zwcJEpU7Sd773Ho/R1A/nkE+47bhwf8Pf1g0hk7lwbm82p3zE3f
k3M99933TZqlHebnqGqImPHou+//4PljD8JYtevd8a1d2RXl55mP60i0jclJYubJpyUJEQq6+mvfo
P0FPdbIgInr5Zde4K6/Uro+/X/Qr874TYyNEJIKZLz52jKe++SZPveoqnnrbbbzqmmt4Z//+b0vc2fv
1qC8WLLBum0zpv5SU5JwYSu3LU0/J8kDL6imKSFwc0ydP5jdffZwniojfKw2LhfvecQcnjxjBRv3r7R
lt2oh8+wW03WZLf1Dfhqcls/hVf7NC6fjs2PPPZ+u0abzmzTd5aqCFgDMzeeAtt/BfBg/mskBrTnTpI
rJzZ3aF3Z5+m76Nh1q/3nX70KHebevPfcoUd1Fwlu2P85XEV7KIiBDZuLHZrenbDx6tPmfduVb4GSx
YoWpsCnJ4rnnZJGvN8PpSBZERA8/XJV0ySXOEiLzs8S7iXh3TAwfHTZMZOTImkvj27dv2PoI/uKGG+y
Sk5NWYDjl1LswroiEXxaZ9mFDjpeYKJKYYeEvu453r1jBu2fPlj9PmsT3PP4477riCt6dmMgVDVmYZ+
lSc6XFkrqtsDDwJzczR11yiXytr68PVa153RIT+eC8eTXnNnUq3zdzJi95/HHenZTEu887jujorzre
oaiiPztbyU2qzXl3dLS0p06J/Lqq67bBq70PoZnGAwi8+ZVF4vY/hhXGL6Sxa0PlrotltQfTaaCvvr9
W47/ZDFhgkt+/jm7yGLJaNSNR50Ju44d63sVrd0VLKimJyJZ3zvTXHHLLRZXTk56Ywlp+j364/pjMnH
fGTP8L1LTnDF/vlWKilJTS0qyGjTu4NAhHjFlhZw1azmirvvrhCTKfVASUn2MP15NMWLL7puu+AC7+
N4RkiIyI03un9pzodFtRh9spg61V5dUJB+1Gz0vEK/b8vynSymT7dXp6ZmmMzmt0cae5Uzfz6/4m8dy
NOZLEQk+09/197s1s37vJoaHTqIPPPusqcpoTD1kNqcv0h+zPj/9xJMvu0xr0YQxe7aNs7Mz8szmjEat
cPXZZ+7rLrmkZRPg/PkVXF1YdsxozLx0f/yTsWyz67bzz/c+nmccv2HbgIccnWaeyaJvX0327cs0W61
pzdKP3TjeyeLmm61Vubnp6Wl6Q81Nv0+9x6PT0jw3zV20pMFHe9Svesu7c1ATz1rSKiqyLRpVfLFF3
lLFkvK58XFmU1e2/GXX6onT5+ulQW639DUWLTIirm5GXnl5RmL9cdtiC++cF53xRWavSFflxoTBoPIf

feVS35+2jGTqXG9Hw21ZInrjt69vY/tGW3biixY4Kyz1soZZ+FCWR4WJtKqFct77+WX2e1pr+xk5NSZ
BXhq/J4sgoJEliwpcxqN6YctlgyfD5oNhJlDJ03i3f663uh4T8/pTBZ0PGFccUX1m4HuzPuL4GCRCRM
csm5dcWVBQdqBysqMR6xWa8B7AA2R1laVMGWK64NAa5o2Jrp1E3nhhRK70Zh2xGRq/0/S0759zv4TJz
r/16mT93GaEnFxmvrX0ab2Zy632z08tm93FzuumN9e31d3j17iqxe7Twt09kbZN48Wd6qlciyZua22
VK/MpvzTsGMB184gYg5IkJk5UpTtdmc+mtFRdMWNL21ipd060D9y/CMe++1is0R8Yi+7qlWkzC0rwMl
Ns+Ijxf5v/+zalu2FFTm5aXvczpTl5pMx/rp2z0ZIjsMy5dXPpaU5LLr72c1NNq0Ebn5Zou2c2e00Wp
N/cBozPY7B6MxRPZGPP20/bmxY53VkJHex21Id0ggsmhRubZ3b3aRxZLyn/p6jJrL3Lmu27t08T4fz7
jgApFVqxyT9HXPCDfeyC//6U9uW1FR+q9lZbl+J/y0tJAQtOyNFVmzxui2WFJ3Fxcfa9IzFY4d49iRI
wMvZBsX5+aff84ywq3pN+jrnw4iMuj667Vyx5f/nTuLXHwxy8KFVlm7ttj2yy/ZaRUVaVtcrrTF5eUp
fudcNIfs0uKBK1eWvT1jRpW9T5+arzz68/MMg0Fk0CCWu+8u5y++yCsxm90+sdvTbzW1wFPBc30LRj3
zTPkHU6ZUORMS6j+3kBCRoUNZkpPLtB07cozl5wlfvlskz8/P998V3BJmznTd3rGj9/l5RmKiZtywwX
7SCazzB0f94x88a8yYip49epSwtw8f/7q+/FSZ0lV6DBtW+ad77skb4XZH/aVDhy4/6/dpiPnz5Y1Nm
+j//E2uUhSi1auN1QsXWjabzfELe/RQm23dh0xf7/rkuRk9am2bR0Xd+2qUffubi02tqoqNpaLevfm
grZtHYciIgyHiegHot5pqqr6+QmbH3PmwG++aT378GGZnJMTkpiToxrKy0nVNFEjIog6dRJJSHBz796
VpRdcUG3q1q16b1CQ7NC06G1RUbFGfxvNiTlnxPbtYbPS09Urjx0LScjJUYIqKhRVp0bcYmNF4uNdHB
9fZb7gAs3YrZtrT1CQusPhCNveunU3v7NdW1JSknuRw+F+NDqae2iaJiJ1Zw0HhYWqM2facmbNsg85m
XNs9mRBxy/t0tKGa0eff+r+AH0WRNcUjIjrmPHjqn6sobYtImvf0opZVtKir7kdxMmVGobNhT+FhER
cktMTI8f9eWhG3Pq7cxKuKoaCoiqsonEpChdShSlTbL+31NNZG+ESFgvovB4Ii3e7aZoTRM1KEh1Ggy
qkUh+I3IXKEriMX/T4VsK8/dRRG17Kkp4ggjHuVzV0UQqGQzkDAoKLhJxpxAFFahqb7+zSE8l5tTB1Z
Xu0YqiaKqq1llJy+0m1WxWQzRNXRcfHx/gQZDQJMwcPn487w50tzwmRuSzz3Is1ZVpj+rrA8A54vnne
Wl9NzXvuafCYbGk7igqKvK5hgEA0V+/VV61XdTMzHRLYcPZ+ba7Rk+HngLcPbBS1k+fPqpRBiN/pf0
V1Wi++83u+Pi3J9t3dr7v/pygLMRkoUPLpfCEuB22uTJldr06RVHnc6Il6+9VvFatBbgbIRk4Ye/ZNG
hA9H995srIyPVDW3adNunLwc4WyFZ+ODvuQ9ERLfcUu4aPty5y+1uf8Y9fhCgJSFZ+OB01jwAWG/Q1L
csWFBmZFb+FhMT43NJd4CzFZKFH/qvIcHBRMnJZmfXrtpH0dHx2+qWApz9kCx8qKryXiN09uwqmjTJd
tjtjvyHopyeZ00CnE5IFj5UVipZzDSY6PeYPbt0cNu2VT0io7s02x03AQAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
X0AAAAASUVORK5CYII=\}"}

也可以采用超链接的形式分享图片或视频内容，客户端识别到link字段则解析（编写对应逻辑）：

代码块

```
1  {
2      "conversationId": 1,
3      "msgType": 3, // 视频
4      "content": "{ \"link\": \"http://vjs.zencdn.net/v/oceans.mp4\" }"
5 }
```

TODO也可以用JSON方便地解析。根据原数据结构：

代码块

```
1 data class TodoTask(
```

```
2     val id: Long,           // 任务唯一 id
3     val title: String,      // 任务标题 (主题)
4     val description: String, // 任务描述
5     val deadline: String?, // 截止日期, 格式如 "2025-12-31", 可为空
6     val type: TodoType,     // 任务类型
7     val isCompleted: Boolean // 是否完成
8 )
```

在JSON中定义对应字段。举例：

代码块

```
1 {
2   "conversationId": 1,
3   "msgType": 5,
4   "content": "{\"id\":1001,\"title\":\"完成项目周报\",\"description\":\"整理本周工
作进展并提交给主管\",\"deadline\":\"2025-12-
15\"},\"type\":\"WORK\",\"isCompleted\":false}"
5 }
```

正常收到响应消息：

代码块

```
1 {
2   "code": 0,
3   "msg": "success",
4   "data": {
5     "messageId": 49,
6     "conversationId": 1,
7     "senderId": 1001,
8     "seq": 4,
9     "quoteId": null,
10    "msgType": 5,
11    "content": "{\"id\":1001,\"title\":\"完成项目周报\",\"description\":\"整
理本周工作进展并提交给主管\",\"deadline\":\"2025-12-
15\"},\"type\":\"WORK\",\"isCompleted\":false}",
12    "mentions": null,
13    "isRevoked": null,
14    "createdTime": "2025-12-08T00:02:57.7387846"
15  }
16 }
```

6 查看会话列表 (HTTP)

假设 LiSi 刚才没看手机（虽然WebSocket收到了，但他想打开App看列表）。

- 接口: GET <http://localhost:8081/api/conversations/list>
- Headers:
 - 无登录: X-User-Id: 1001
 - 登录: Headers: `Authorization` Bearer eyJhbGci...iJ9.eyJd...MDF9.TY4d...m08
- Params: 无需任何参数。
- 预期:

代码块

```
1  {
2      "code": 0,
3      "msg": "success",
4      "data": [
5          {
6              "conversationId": 1,
7              "type": 2,
8              "name": "周末去哪玩",
9              "avatarUrl": null,
10             "lastMsgContent": "Hello",
11             "lastMsgTime": "2025-11-29T21:10:19",
12             "unreadCount": 3
13         },
14         {
15             "conversationId": 2,
16             "type": 2,
17             "name": "周末去哪玩",
18             "avatarUrl": null,
19             "lastMsgContent": "ZhangSan 邀请 LiSi 加入了群聊",
20             "lastMsgTime": "2025-11-29T21:09:33",
21             "unreadCount": 1
22         }
23     ]
24 }
```

7 查看历史消息 (HTTP)

假设 LiSi 刚才没看手机（虽然WebSocket收到了，但他想打开App看列表）。

- 接口: GET <http://localhost:8081/api/messages-sync>
- Headers:
 - 无登录: X-User-Id: 1001
 - 登录: Headers: `Authorization` Bearer eyJhbGci...iJ9eyJzd...MDF9.TY4d...m08
- Params: conversationId: 10 (替换为群聊ID)。
- 预期:

代码块

```
1  {
2      "code": 0,
3      "msg": "success",
4      "data": [
5          {
6              "messageId": 32,
7              "conversationId": 12,
8              "senderId": 1001,
9              "seq": 1,
10             "quoteId": 0,
11             "msgType": 1,
12             "content": "{\"text\": \"ZhangSan 邀请 LisI 加入了群聊\"}",
13             "mentions": null,
14             "isRevoked": 0,
15             "createdTime": "2025-12-02T19:46:23"
16         },
17         {
18             "messageId": 33,
19             "conversationId": 12,
20             "senderId": 1001,
21             "seq": 2,
22             "quoteId": 0,
23             "msgType": 1,
24             "content": "{\"text\": \"Hi, world!\"}",
25             "mentions": null,
26             "isRevoked": 0,
27             "createdTime": "2025-12-02T19:46:45"
28         },
29         {
30             "messageId": 34,
```

```
31         "conversationId": 12,
32         "senderId": 1002,
33         "seq": 3,
34         "quoteId": 0,
35         "msgType": 1,
36         "content": "{\"text\": \"Hello, world!\"}",
37         "mentions": null,
38         "isRevoked": 0,
39         "createdTime": "2025-12-02T19:46:57"
40     },
41     {
42         "messageId": 40,
43         "conversationId": 12,
44         "senderId": 1002,
45         "seq": 4,
46         "quoteId": 0,
47         "msgType": 1,
48         "content": "{\"text\": \"G1\"}",
49         "mentions": null,
50         "isRevoked": 0,
51         "createdTime": "2025-12-05T21:42:51"
52     },
53     {
54         "messageId": 41,
55         "conversationId": 12,
56         "senderId": 1002,
57         "seq": 5,
58         "quoteId": 0,
59         "msgType": 1,
60         "content": "{\"text\": \"G2\"}",
61         "mentions": null,
62         "isRevoked": 0,
63         "createdTime": "2025-12-05T21:43:13"
64     },
65     {
66         "messageId": 42,
67         "conversationId": 12,
68         "senderId": 1002,
69         "seq": 6,
70         "quoteId": 0,
71         "msgType": 1,
72         "content": "{\"text\": \"G3\"}",
73         "mentions": null,
74         "isRevoked": 0,
75         "createdTime": "2025-12-05T21:43:16"
76     },
77     {
```

```
78         "messageId": 43,
79         "conversationId": 12,
80         "senderId": 1001,
81         "seq": 7,
82         "quoteId": 0,
83         "msgType": 1,
84         "content": "{\"text\": \"G4\"}",
85         "mentions": null,
86         "isRevoked": 0,
87         "createdTime": "2025-12-05T21:43:23"
88     }
89 ]
90 }
```

8 查询用户列表

客户端启动时自动调用，以获取用户ID、用户名、头像等信息。

- 接口: GET <http://localhost:8081/api/users/list>
- Headers:
 - 无登录: X-User-Id: 1001
 - 登录: Headers: `Authorization` Bearer eyJhbGci...iJ9eyJzd...MDF9.TY4d...m08
- 预期:

代码块

```
1  {
2      "code": 0,
3      "msg": "success",
4      "data": [
5          {
6              "userId": 1001,
7              "username": "ZhangSan",
8              "avatarUrl": "https://api.dicebear.com/7.x/avatars/svg?seed=Zhang"
9          },
10         {
11             "userId": 1002,
12             "username": "LiSi",
13             "avatarUrl": "https://api.dicebear.com/7.x/avatars/svg?seed=Li"
14         },
15         {
16             "userId": 1003,
```

```
17         "username": "WangWu",
18         "avatarUrl": "https://api.dicebear.com/7.x/avataaars/svg?seed=Wang"
19     }
20 ]
21 }
```