

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                not null,
6      user_id            bigint                not null,
7      last_ack_seq       bigint                default 0                null comment '已确认同步
到的序列号',
8      unread_count       int                    default 0                null comment '未读消息
数',
9      role                tinyint                default 1                null comment '角色：1=成
员，2=管理员',
10     is_muted            tinyint                default 0                null comment '免打扰',
11     is_pinned           tinyint                default 0                null comment '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)            null comment '群名
称',
24     avatar_url          varchar(255)            null comment '群头
像',
25     owner_id           bigint                default 0                null comment '群主
ID',
```

```

26     current_seq      bigint      default 0              null comment '当前会
话最新序列号',
27     last_msg_content varchar(512) default ''          null comment '最新消
息摘要',
28     last_msg_time    timestamp    default CURRENT_TIMESTAMP null comment '最新消
息时间',
29     created_time     timestamp    default CURRENT_TIMESTAMP null
30 )
31     comment '会话表';
32
33 create table message_reactions
34 (
35     id                bigint auto_increment
36     primary key,
37     message_id        bigint              not null comment '关联消息
ID',
38     user_id          bigint              not null comment '操作用户
ID',
39     reaction_type     varchar(32)         not null comment '表情代
码, 如: thumbsup, heart',
40     created_time     timestamp default CURRENT_TIMESTAMP null,
41     constraint uk_msg_user_react
42         unique (message_id, user_id, reaction_type)
43 )
44     comment '消息表情回应表';
45
46 create table messages
47 (
48     message_id        bigint auto_increment comment '全局唯一消息ID'
49     primary key,
50     conversation_id   bigint              not null,
51     sender_id         bigint              not null,
52     seq              bigint              not null comment '会话内
序列号',
53     quote_id         bigint      default 0              null comment '引用的消息
ID (若不为0, 则是回复消息)',
54     msg_type         tinyint              not null comment
'1=Text, 2=Image, 3=Video, 4=File, 5=TodoCard, 6=RichPost',
55     content          json                not null comment '消息内
容载体',
56     mentions         json                null comment '被@用户ID
列表',
57     is_revoked       tinyint      default 0              null comment '是否已撤
回',
58     created_time     timestamp default CURRENT_TIMESTAMP null,
59     constraint uk_conv_seq
60         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/avataaars/svg?seed=Zhang',
15  NULL),
16 (1002, 'LiSi', 'https://api.dicebear.com/7.x/avataaars/svg?seed=Li', NULL),
17 (1003, 'WangWu', 'https://api.dicebear.com/7.x/avataaars/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/avataaars/svg?seed=Zhang",
9              "password": null,
10             "createTime": "2025-11-29T17:03:34"
11         },
12         "token":
13             "eyJhbGciOiJIUzI1NiJ9.eyJzdWIiOiIxMDAxIiwiaWF0IjoxNzY0NDExMzAxLCJleHAiOiJlE3NjUwMTYxMDF9.TY4dhmnW5Q3AhL0fGMVkhG0xLiraofv9Ucnh7Unzm08"
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`
`eyJhbGciOiJIUzI1NiIsInR5cCI6IkpzdmEiLCJmYWxzciI6IjE5MDYyMjY0dC5m08`

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`
`eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXZWQ6MDF9.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 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpzdmU6MDF9.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     "createTime": "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  }
```

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：

```
1  {
2      "conversationId": 1,
3      "msgType": 2 // 图片
4      "content": "{\\"base64\\":
    \\"
    B2cksfwAAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAACxEAAASRAX9kX5EAAAGHAvVRYdFhNTDpj20uYw
    RvYmUueG1wAAAAA8P3hwYWNrZXQgYmVnaW49J++7vycgaWQ9J1c1TTBNcENlaGlIenJlU3p0VGN6a
    2M5ZCc/Pg0KPHg6eG1wbWV0YSB4bWxuczp4PSJhZG9iZTpuczp4PSJhZG9iZTpuczp4PSJhZG9iZ
    cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOkRlc2N
    yaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTnkLTExZGEtYWQzM51kMzNkNzUxODJmMW
    IiIHhtbG5zOnRpZmY9Imh0dHA6Ly9ucy5hZG9iZS5jb20vdGlmZi8xLjAvIj48dGlmZjppcmllbnRh
    dGlvbj4xPC90aWZmOk9yaWVudGF0aW9uPjwvcmlRlc2NyaXB0aW9uPjwvcmlRlc2NyaXB0aW9uPjw
    bWV0YT4NCjw/eHBhY2tldCB1bmQ9J3cnPz4sLjJgLAANA0LEQVR4Xu3dd3gU5fYH8D0TTSchdEIPiQY
    Q6V0UA1goIiDYAH/iBSzYI3rVawWVouIt3iteEQVBL6jXhijqxUpRVDqmk55sd102JLubnXN+f4TgZr
```

```
1  {
2      "conversationId": 1,
3      "msgType": 2 // 图片
4      "content": "{\\"base64\\":
    \\"
    B2cksfwAAAARnQU1BAACxjwv8YQUAAAAJcEhZcwAACxEAAASRAX9kX5EAAAGHAvVRYdFhNTDpj20uYw
    RvYmUueG1wAAAAA8P3hwYWNrZXQgYmVnaW49J++7vycgaWQ9J1c1TTBNcENlaGlIenJlU3p0VGN6a
    2M5ZCc/Pg0KPHg6eG1wbWV0YSB4bWxuczp4PSJhZG9iZTpuczp4PSJhZG9iZTpuczp4PSJhZG9iZ
    cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOkRlc2N
    yaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTnkLTExZGEtYWQzM51kMzNkNzUxODJmMW
    IiIHhtbG5zOnRpZmY9Imh0dHA6Ly9ucy5hZG9iZS5jb20vdGlmZi8xLjAvIj48dGlmZjppcmllbnRh
    dGlvbj4xPC90aWZmOk9yaWVudGF0aW9uPjwvcmlRlc2NyaXB0aW9uPjwvcmlRlc2NyaXB0aW9uPjw
    bWV0YT4NCjw/eHBhY2tldCB1bmQ9J3cnPz4sLjJgLAANA0LEQVR4Xu3dd3gU5fYH8D0TTSchdEIPiQY
    Q6V0UA1goIiDYAH/iBSzYI3rVawWVouIt3iteEQVBL6jXhijqxUpRVDqmk55sd102JLubnXN+f4TgZr
```

akkADC9/M85+Fh3nnfmWyyZ2fnLUMEAAA
AAA
AAA
AAA
AAAAAAAAAAAAAAZzxFv+FkLRLp8bZIV6u+oOU5SFV/1W88K4i0IpEL9ZtrhRFRWyLqSkRjiGh2zb95q
qrm6vcFOGMSZH4hRkToFEawiCyw2z835uWdrz+fswLzaP3P7BmqiISKSLSI9BCRYSIyGLmwN/PHi5m3
3M48JZM5Ud8swGm1gHlFax9/0C0Zd5aXc0lq6qGCrKxJ+vM5KzCP1P/MDQn1eCKNEZG+zNaJzB/ /mXk
OM3fQHwLglFsosvxUJosZNpsrPzU1zZyWNrv0iYiEEfMYXxHGPIZEwtfZ/0zWxGShjyARaS8iI5kz7x
N5Eknj9HqJuZN+2znLVCaLUQ6HMzUj45g1Pf0e/XkQ83n6/WsjREQmVVUt46NHo/TVzkjNlCw8I1pER
jNnPMo8R0RU/SGh5Z3P/NBo5nVy6FCivuxM9If9I0lkpr8VFV6kMf+nVXz8y/pyIhL9hlouIppgty8i
RRmgLztXWIho16L0Xq8ob81nflEgVX7QMtKJararyg3z+7b9/Ny5gR9+ZmD5ks0orQiwUF1r4u19b
c60iliqK49fsQEZEI6zfVEiI3MfstP1fkEtE7qnrtUJF3D4m00pdDi9IqiegTVb10rsj/vnU4+ul30J
P84ZJF0BE9YzSWj6uq+p+pdes/9+/Uyabfp4H8XnmcaxxEdFBRps0VeWcvrjB00QcRbVXV7ncZDJ8/7
HSesVe7zT70YqHI8s1ED1XoC4iok6a5HzaZcs0ZQ7UmHrsDc9DYqqPMNSzstk7du+/Xl59Qc1mXQori
MyGuMpsLko3GWWq/frv0ZWcc5pGkKLv1m4mIQono7tLS4v0cztDykJDWJo0B0g0Gyo2MVHKIqFhfiYB
QIrpK097fkps7R4mLc+jLoZkxLyZF+Yfnpp4iJcLED9ytqus9t58JmvSGDSRQskjUNP4+K+vDdnb7Cn
1Zg4kYsqKjS+Lj41P1RXWcI8kioi2FBYevNxoFJQUYRCQ4PIYOiUGxo66Mfg4D57w8JG7GzduucvR
GTxV/ahGxEtrqy88+HISF/3gaA5+UgWRET9iWgm8/KnVPURRVH03ivgQL0hicxyNDt7g750i2B0IGZN
fw61scpkycujR0brq52RAvSGRInIu0bjfv7lF5/doPzrr10LcnNnLS0v3zLG7ZYgh23oY7zLVcwFBT3
1bUEzY16sf+1r4yIROWIyjdJXOZ18fuq2qJru/ma/ojmXiYiIH6rcTEamDB+d36d793cdiYmavNh
rHTXI6d9TXX3w40LjjW6Gh9+m3w6mjERE5HPP120+nU58s4LQZ2aXLtx8XfK6+WtM+CdcXeigmoimGw
/W8b19HfRmcGgoRqSJn1PgLJIitzjBIX59gQFDR7oMhhfZmnlKCgDhQRMUi/HU6pM6prH8niHKQoiqML
0YpQfYGHYKVRHKO6WL8dzl1IFueoXUSHA13jaqqVDF30W+HcxeSxTlqEtGCSv1GD9HMQiIG/XY4dyF
ZnI00MffbT3STpi/w0E/TLIigoHL9dj3IVmcY0Sk9XSilfsUxWdXKx2/Ez/EzisPVdUD+jI4dyFZnE
M2Ml81XmT3EUWZEuYqYoCm0RiHo8BFtE9fBucuJIs/ODl+JSAul8+ZtyIS8pzIxAnMG5cQvb9DUfq49
DvpzCstdcVWV3+7Ky4uQ18G565mTxarIk5/wzNVImrvdlef6ePds5j7WltwFaP/E+kTypxE/m0IrkqQ
7v8naEQUzswUHBzBzEMOMic9zJw0TWT2rcyf9hHZ82+iL/6nKDcWKEq9NyyTqqr42oqKTEdIyJtJ/qb
+NxNm7vQuc9J45qR2zEmhzEk9mZnuZE76njLJRPz+3C2Nmbu9xZx0CXNS6+PnFs+cdMvxc9Pvfzq9zH
z/9cxfxSH85SQfMzn5q8eZT3pErr/3tV8PMvd9iaixQ2ZmrZeLiRb+oig3+Jq01F2E7jGZ/vdQ+/bPa
CcxRTxWUTIKFSVHv720Jk4kE5GgPxHt0ZgtbwYFj fcsaw5m5u7TiA7tUpRofyNu7nM4fnsyP39a64SE
NCIiYo6eYrNdP8NqfVpq3sBCRBSpadYDkZFtt0VGRpYaDDmdFeV8RVEMRiIqJKLqRo7q6a1p8lZ+vrL
/dfXfoxMSntaXNwcR6fwo0bTfRK7dR5QqoSg984nIdjzxhRBRzPGVys1EP48RyeLfT04ZRfLUUZRA35
50mpm5+/NEMw8RXXWEqG+YonSpPTc+Piu3DRF1JqIykd0XE2V3I1r3rKp+rm+rQfxMJCMiGk1E6/LyX
u/Tvfu9GWe3mB+4CVFWeXv5hIT0QQiWmk0ru3HvCw8NvaYfp8W83fmYZM0zRTsY/LLqYrpXPkhf+t7
4tQJ9U0kqzDl8wHviWT/ZZ7QS0Rai8jjbLVX36yrmX+R6iP86mNEU4HZ2VkJh9LD7MBw4MkwMHhvL
+/UM4a1ffffC5+aCy7XPo2mhI93W5ta16e0ZKauv5oSUL900cajZk73c28eiBzdtvj iwnrz8FXqCLstW
bF8r33MN/SELcbOcxdb2f+Z3/mgtanPLduIjKcedcS5jn6dusVYCLZaBH5LTd3rb6Kp/8w3znMR13PG
OJwuvZnZeuUpqauPHqSy0g2+sqCiCivouL8J5WIrVtSDQkVZbovMw25XmjSUX/3RCcrPRGRfb3a+8L/
6Mt02MEJ5A5wZdHRXJACZZylxv1+ZSEiQTftlC83VSmXairRUHY7nnfbL0y6Mia9bu2mee0nTlpqoe3
Z4vvrQIiBaH3vQstVLttrEFHnPRDo61pBAQ+/7ih9/J2idGrU5YMPw+KrtTUdC/MH2+zb7BFtHo+Kij
Xq92kqEYL49i2Z99/W9PBvnZWeturjq734/cl8UIioFVE7IhqSL7/8XyU9MHeuuk0/W20JSMYj6+XmT
9vRktROSPdKJxEm3BuUUTtmWhkvuy5men02bPVvfrdFPqSfXP5ubJoTbSuQ97rFxr5vrL47ze84EkX
vbo/SFF8/v6dRAN6u2ltdF5hYqV7szkh4c9xiuLz20CL4/fs3W64XFsfRSJ0imPy5XCf0Ba/TnVxQL
ErOnr1saqZ035zHWvLD75RCYmnvf7PiQJzJxZvUsku43nfk3BzKETL+M9qo9zqY0502xszkvbY6zIPU
9f350IhE2dyimKjzaaEnHtq90P32uzbdjoWNKcn9wiEnHHHfxZr17ex2xqKKpIXBxbVq/ml09moeGsL
IlZtIh/6NrV+xhNjSCDSGIim195hVfpj+cbL9a3URujB4r89pvvK4vPP+dbL77Yu45n90nG8sNn2XZr
cdprqSZTtL6N02LG1dXPdon1PtmWjKeftovI4bpL/3upJ1msqpssRCToppt4R1BQ3f06dRJJTq56rm7

bjbd6Na9o1877PE4cp6PG03/INTnt6bfr63oSkeDLL+ctBoN3GycbnTqJJCXxgRUr+BYRCdMfuzF27J
BWo0bxZ2Fh3sdpjoiLE3n0Ud4g8kSjE8Y//ylthg/nH0JCvNttjkhMFHniep/6o/rLUCyG0U7Wwzdy
pcNH84u/f6e0bMHY5df5Nit1vQ3c3LK2+rb0K2WLHE+26eP90m3VLREsvjoI57k72dITHRXv/tu2fS6
7TdCtG53HTKEK/TtesZjj5U6bLbU/5aUsN/vlcwc+sADvKVtW+/6zRldu4rMmsU/b9/OE/Tn0BA50Rw
+eDBvDw72brs5IzZWZnmy6g2NWZl83TqJGTCA7ZESvWMnj1FVq1y/jPw1U+AZDHa01ls2cKDBwzgIv
2+ntGpk8iHH+bZrdaUzenphWfmEgNr1lQtHTbM++RbIloiWVxzDb+rqt771cb48dVpzHnt6h6jfiISP
G8ebwr0KTZ0aLWkp2ccs1qPBXxzrl7N/+je3bt+S8UFF7Dj4Yd5cW0uMpg5auZM3hYa6t1eS0Tv3iJP
P+28RX8evjBz20mTWu6KQh99+oisW0EI8CHT8GSxfLn06NeP0/X7eUabNiJvv11gs1pTPkpJyela91h
nmPXrK2f36KFLBXRtNUc8+aSj2ZPFihXyZFyc9361ERYmkpxs3173GPV74w2e5u+KhUgkKEhk/fpCa1
VV2kuBVhBbuJAndenC1fr6wcEiF15YVXXRRtB3mDF2bezYSpk4UWT8eJFRo0T69RNP3dr7uA2N+HiR5
cv5I5GsBiWMRx7hVe3be7fjKy68kGX+fKusWGFyr11b6HrjjTzLa68VuZYtK3XeeKNNBg+ueX309fTR
v79W8e0P5jH6c9G75x5+PTrau76vGDRIk4ULLfL88yXutWsLnW++mWf597+LXE89Veq87jq7DBggoij
e9fQxbFh1cU50kZ+HWgdOFke05L50RLRsGcf27cuHA72vIiNF1q4ttlitKZ9mZ2f31h+p0fj942yqBQ
v46c2blccsFn0JUef0mvuJJ4zZ4eEcrGLKk4+dmMitevSovKNHj8R39GW/Oz70gvz0hqwyFyQnG2epa
k1viIgYZs3iVz75RP2T06nfu0aXLsL332+79YEHol/TL/nCzKFJSftjN9/4f5jRNddU8po1BftDQiLn
REd30aovJyL6y1+473vv0d6jR5UIfVnHjktVv59/aPjwim8rK4M0IkwtrLZauVyqWl4eRGVlocEFBwp
0fr6h/4EDod00H26LHjLCZPc1EMaPzp2J7r1X+/jBBzNnq+r5fl4doo8+4iEPPkg//PabEjCx90lTzY
sXV1Refrm1qEsX187Iy0AFfYX2E1lsRMEkIkIU2vHIkchLNm6MGL91a8yYA/4GEhCrWUA0b57j9p9dfz
xirKP19DLBdu5bHPfMMfZmZ6bsnqtagQdXaHXeUVk2YUJnXsWP1zsjiOB+J5KCi2Cp/P7fwLv2RYzb
tCl8wtatMc00+vyt1QgPJ5o/v/KLl19eeawiPKXrtwjQGzKaaMuWY3/Ny0i54q676LMjR5QBbj9D5IK
DiVatMLXMn1+6r7IybHFsbM+ACxudMW69VZb7+yRLTNQ0ozHjPZFFB51M200Hh//yS2rgcRaNVlKgmj
d3u6uvZn0gDH7RRS7bwYPm7p71/HnySV4RE+PdRm106CDy7be5ZRUVv92vr+vJY0Ax+rq1ERsrm1b4
Q+BviiqIHAPHtyT+dCI1NT8B19+2bx91iyHtTFfaXr0EHngAffN+rY9XXYZ/6e+T9sZM+xy8GBWgcOR
stblyhihb80X774zXnvxxa7DgW6WxseLPCBcaG+bq1x43h7fec2d65VULIycx20lH/abFn1rhIm8oR
h2zbTzSNHammBvtr07y/y5ZfFs/T1A11ZjBnj1jIy8t4fM4YPBLq/oqoiy5aZrRZLyq7i4rx6r670KA
sXBkoWLEeOZL2lr9MyGp8sq0Zrw4j4eLbq96+NoCCRG2907BIpDvj0rv370fGCCwLf1HzooTKn1Zrya
VLZVoy+fL08UL+3NmJjRT79tHBnY7t3mVP6btXy8ey0aa68QAnNMwYM00zfmsaqW+Li0ivf+XBPXuy
Q1/HM665xi55eel5FRVpyVLZ/p0bLyULJVHDh2tFBUrkixbZ0ph3e3UTLl3Kl3Ttym79/p5x881WKS5
OyyovT71t796G3zAlIvrii4p2gwZpe/wlo6Agkfvs/zK/Ffd4mT+k0X//i7t6qurONDPSyTyl7+USl
lZ6sHi4uzL67b9B1Bfsjh69NimQN/Nm0/TkgUR0Ysv8lOxAbqDY2NFHn/c9oS+Xi0RCZ47lz809GkzY
EC1pKRk5lRUpe/R1/dWf7JgzmlSFxlzSteHHrK/1b0nd9v6UFWRefPs34rs8Hqj3347bwp0f6Ffv2rt
yJHMFks1/V593YbKz0TEES04Ut92bYwaJbJvX+5N+no33cTbAp3b80EOLSSrM7usLG2Bvm5Ddf89Dxs
40H+XZlKSSHZ23lV1a/LPFv4Sj2fce2+5mEypKWbzsbQb8DZ2BzoZkISLq90n8dqA7+oMGae7PPiu/TF
+XapLN1YFuLqqqyOuvF7oqK9P+tXlZQwZBtVyyqHXnnc7rOndmi759fVx4oci0HSVzPesyc9uhQzlNv
29tBAWJrFtBHM4UL8hOrnf/V138f3+uo4jIkRef934hefAMmbu2L8/F+j3rY2QEJFNmwsTmfq83WP
1HgLFmjL/N1AbdNGZPPmonfr/u37Txb1xYIFFWw0pqebzZn130g/g50NyYJqEkaviRPZpK/ngVdcUf2
r/k0qIjETJnCwfl/PmDq10m00pu+zWnP83CXXa/lkQUSUnOxeVN+IRlUVWbq0bLfI3hOX6l9/zZMHDf
LetzYGDnRpWmZB63WnP51j9h4zBwVF+c/MSUnV5SKH0pRu/8HH/CcQL1RF13kcBcWZvxUVXUsru6RG
q+ggDt07+4/MT32WHku8xce3e9NSxbXX2/VCgrSM0tLMxo/H+Uk+OwpACJFUY5NnkWzzj9fX/K7XbsM
gx56qMNFbPbddd5888tNPSi/PbZ7atye6915TZUQEbYyK6nFQX346Pf980PqLL5av9Ns9MRMDPBg8VCR
yY0225csp6FiAuYyXXmqn2Njq76KiehzSlzWWqqrWyy8nvw/DzsgIiRQJG1r7/xdeIEN+ft19PE2YYJ
e0Hd1fhYf3ytKXNVaXLMrJuHFUpd9eKzPT0EZReniumN7oD80pUyq1FSuKc0ND5dm2beM36stbEpJFA
Pffr343b56sjvFz+9FiIXrnndAb3nijbAER0Z49PHrrVrrNV7dxrZtuKq8eMcLxY2lpiNdQ3tNNURTH
rFn04nkBZ6YQFRZGGAoKwk98Ty4vp+LISPE1nYmIiAYOdFaFhp78xK9aRUX0psFPB6jTGRZiNkrDa//
PTAWt/NyKVhSiAQMcVkvRv9aXNVVBaw30d27V1ZGRdnvwiWSRkEBBoYGex6Bz6aW06hdfLCps25Zeet
fuvAZ13zcnJIt6PPaYev/ll8vXYV639Grk5ipBr7/eamVZWVxcSy/R09nZit/h2n36aLRgQVkuFDLP
Xr0KNWXnwmuuUb5VLHkF/12TzabQoWFakLt/3fvVn+88kptdp8+7DWrMSyMqF07d7GI59n6/ouLKSPK

z6tcVUVUUA Gc6BH54Qf1i6QkuSkhQfNKZq1bE7VtqxUQWQKMLGic8nILM8JrNEyNykois/n3c3vwQXp
v6FD5KSTQMxmOGznSKa+9VpjbPQuvff75+L/py08FJIsGePVVenDIEC7Rb6/13XeGNosXKz9u365MdP
kcElTzKXbPPaXVvXq5t/38c6+P9eVnChVRbJGRVBJoyJzTSeR2G1p53ntauzb4/eRkx7WXXqo5P0tGR
RGFhytWoLDr71tPTmwwshQb5uS3sdh0JqHWezrhpk7rh9tur5o0ZUzdhtG5NFBamVChKZC0GqAXWubOE
qn7eVW43kaL8PrBu0SI1789/tk3u10/70biezLqHI0jJyQnLjoo678mnnlK8Et+p40fHAK8xMepPU6Y
oc/r00ZFUECEqKjK0Hzf07+BGuuyyKm3GjIoUpzPyLaSkRi9XF+Ct2/xsNvK78jcdHzEYFuY01S/3t3
Bh5Mf/+pfz2iLTJLP2j1+Ey0VSrIqi+f0u31gHDPDL4XUNUyM0LCgszPt5J8nJUztWrqy6afx4zqtNN
CJEmmYoJ3L7aa3xULPJ5e8DIzycKCys7hi0ad0iTuUWBF153nm8118CJCLav99Ac+d2HrxihQSYa9Ky
kCwa6JFHlK+mTZNx/H3/jYlxOdasMa2f0JGL9J/KMTFEycml9lat+J02bboGvMT3ZeFCkoQTF/0t6/v
vZZCi0DAJsABMZCRtu3buqtqHfXvq2zfy402blJFjxmgfRUYSmc01yUfE0Wx/a5dcQhNtfm5xtmrllr
ZtXT7vGo0dG7VxxYqqEaNGaTvCwoiKioisVq2VSFCzJeNBg2h8pZ+nN7Vq5ZLWrau9zm30HmV0552VU
xITZb/+b8dTQYES88or8s6zz/JUfdmp0Gy/wL0doi8cqV6+8SJ/JPvS0ZDWL2tVd//+9/qpMmTudTz
lz5njtU5apT9V5HWTboptWYNHbr3XlnfvUGDzE/OunVyX3Z24Lkd3bo5KCamutDfSL7R0Yrp66/3zZo
zx/Vh+/ZE6ekhvRXFELDNxjCb6VL9tld01dyeLhm1m+vNXx4q8Lvvy+eev311V9FRRGl4fEE1kacN
egYSwWuLi/rVanTnZ3cLDb572q0+6IMq5cqwcPFgC9CsRZUUpoWvX0jsPP1x9yhcnRrJop0XLlcX9+
rFJv51IyGYztIuLU/b9/e+OyV0mcCKRUe/emtx6q8mkqsH/iorqXKyv1RCqqtoXL1Zunj2b17dr9CT5
hlu7lmdt307X+ptIV2vEiEpHWJgasKtRUYZVv/pq6PQ77+S1Vmt4uIjaLM9NTUvj0ZmZ5HdW5ZAhDs1
gCA74hl0UrpXr1j1z+cKF7s1GY0g7oshmWcl9zx6+IjOTfLaLKERDhrhcRKHZ+rJaU6cqmcuXu28YPV
q8rj48ZWQokW+/rb67YIHrRBfxH9LZMigrkNWr+YbzPJbfxKZ0dMpFkvmQ7X7pKbaR02erJleftkmd
nvqusb0N/BFRJQpU6o3hIf/ftzmGpT15Zc8a9w4/80Va6NzZ5Fdu7JzmNPqvHYPPcQ3tWrFe4j4h99D
vg405h1DhLS7t2wxL/Xcv6kWL0DnIyK8z6v23HbuPJbDnHJiDAgR0eLFvDgsjHfXPTfeERrK340ZU82
ff24M0JGvoW68kdf6G/Xbu7fI/v1ZKcxp9X6h/OEHmT5uHHu1oY9evbSiu+5y/nGfdH8uJAsRURcs4B
c9Z0HqkwURUVLZVVxm5rFl+fnZYz23nwwRUcaN409q5zqcbLIQkZAnnuA7hgzxXivDV9x2m0WzWFI/L
NGtAP7JJzxywADv/Wvj8sudJubsk7q6+OYbHjNsmP/zvOGGKikrS/1UP8lvwwaeEGU51VXuXJFMlp7
1mms99/nSRdeyKxvuzZuvdUmVmvaZubUBo2s+04716IJE7zb8QxFEULMdBc/99wfNGEESHZxcSKHD//
xkwUR0fz5vLS+ZNFsMLnvjTdkOR1fRq2pyeKFF3jEtGm8tUcP79fHV3TvLrJnzzGz1Zp+vb4tIqIbbp
A3/E2nNhhELr1U+0gk388ohMCYue3o0bzT3wSr4GCRLVsK7C5X2L36ukRE06fzh/4mk4WGikyYoG2SJ
j41/sgrjh02jPfr262NyEiRbdtyLU5nesAp/nrbt7tvq29hXkURGTTIXbr5s7PeKfVnnEDJYtq0asnP
z9RNpmkpLZcsPv2UxyUksN0zvVOZLIiIMjNd48a0lfl27WvXs6hvmnsNZr7wxRd58XXX8c7YWHb7e/P
pQ1FEVq820Wy2tPeKi31Pz9+zhy8YPJht+rq1ERYmMmGCtm/TJvY5Ac+fH3+U3hMn8g+eX7/0cdVVVV
xULPpzVWWhz6H2n3zCI/v2rfs784zwcJEpU7Sd773Ho/R1A/nkE+47bhwf8Pf1g0hk7lwbm82p3zE3f
k3M99933TZqlHebnqGqImPHuou+//4PljD8JYtevdY8a1d2RXl55mP60i0jclJYubJpyUJEQq6+mvfo
P0FPdbIgInr5Zde4K6/Uro+/X/Qr874TYyNEJIKZLz52jKe++SZPveoqnrrbbbzqmmT4Z//+b0vc2fv
1qC8WLLBUm0zpv5SU5JwYSu3LU0/J8kDL6imKSfWcOydp5jdfZwniojfkW2LhfvecQcnjxjBrV3r7R
lt2oh8+WW03WZLf1DfhqclS/hVf7NC6fjs2PPPZ+u0abzmzTd5aqCFgDMzeeAtt/BfBg/mskBrTnTpI
rJzZ3aF3Z5+m76Nhlq/3nX70KHebevPfcoUd1FWlu2P85XEV7KIiBDZuLHIzrenbDx6tPmfduVb4GSx
YoWpsCnJ4rnnZJGvN8PpSBZERA8/XJV0ySXOEiLZS8S7iXh3TAwfHTZMZOTImkVj27dv2PoI/uKGG+y
Sk5NWYDJl1LswroiEXXaZ9mFDjpeYKJKYyEevu453r1jBu2fPlj9PmsT3PP4477riCt6dmMgVDVmYZ+
lSc6XFkrqtsDDwJzczR11yiXytr68Pva153RIT+eC8eTXnNUnUq3zdzJi95/HHenZTEu887jyujorzre
oaiiPztbyU2qzXl3dLS0p06J/Lqq67bBg70PoZnGAwi8+ZVF4vY/hhXGL6Sxa0PlrotltQfTaaCvvr9
W47/ZDFhgkt+/jm7yGLJaNSNR50Ju44d63sVrd0VLKImJyJZ3zvTXHLLRZXTk56Ywlp+j364/pjMnH
fGTP8L1LTnDF/vLWKilJTS0qyGjTu4NAhHjFlhZw1azmirvvrhCTKfVASUn2MP15NMWLL7puu+AC7+
N4RkiIyI03un9pzodFtRh9spg61V5dUJB+1Gz0vEK/b8vynSymT7dXp6ZmmMzmt0cae5Uzfz6/4m8dy
NOZLEQk+09/197s1s37vJoahTqIPPusqcpoTD1kNqcv0h+zPj/9xJMvu0xr0YQxe7aNs7Mz8szmjEat
cPXZZ+7rLrmkZRPg/PkVXFiydsxozLxOf/yTsWyZ67bzz/c+nmccv2HbgIccnWaeyaJvX0327cs0W61
pzdKP3TjeyeLmm61Vubnp6WVl6Q81Nv0+9x6PT0jw3zV20pMFHe9Svesu7c1ATz1rSKiqyLRpVfLFF3
llFkvK58XFmU1e2/GXX6onT5+ulQW639DUWLTIIrm5GXnl5RmL9cdtiC++cF53xRWavSFflxoTBoPIf

feVS35+2jGTqXG9Hw21ZInrjt69vY/tGW3biixY4Kyz1soZZ+FCWR4WJtKqFct77+WX2e1pr+Xk5NSZ
BXhq/J4sgoJELiwpqxqN6YctlgyfD5oNhJlDJ03i3f663uh4T8/pTBZ0PGFccUX1m4HuzPuL4GCRCRM
csm5dcWVBQdqBysqMR6xWa8B7AA2RllaVMGWK64NAa5o2Jrp1E3nhhRK70Zh2xGRq/0/S0759zv4TJz
r/16mT93GaEnFxmvrX0ab2Zy632z08tm93FzuuMN9e31d3j17iqxe7Twt09kbZN48Wd6qlciyZUa22
VK/MpvzTsGMB184gYg5IkJk5UpTtdmc+mtFRdMWNl21ipd060D9y/CMe++1isOR8Yi+7qlWkzC0rwMl
Ns+Ijxf5v/+za1u2FFTM5aXvczpTl5pMx/rp2z0ZIjsMy5dXPPaU5LLr72c1NNq0Ebn5Zou2c2e00Wp
N/cBozPY7B6MxRPZGPP20/bmxY53VkZHex21IdOggsmhRubZ3b3aRxZLyn/p6jJrL3Lmu27t08T4fz7
jgApFVqxyT9HXPCDfeyC//6U9uW1FR+q9lZbl+J/y0tJAQToyNFVmxui2WFJ3Fxcfa9IzFY4d49iRI
wMvZBsX5+aff84yWq3pN+jrnw4iMuj667VyX5f/nTuLXHwxy8KFVlm7ttj2yy/ZaRUVaVtcrrTF5eUp
fudcNIfS0uKBK1eWvT1jRpW9T5+arzz68/MMg0Fk0CCWu+8u5y++yCsxm90+sdvTbzW1wFPBc30LRj3
zTPkHU6ZUORMS6j+3kBCRoUNZkpPLtB07cozl5WlfVlSkz8/P998V3BJmznTd3rGj9/l5RmKiZtywwX
7SCazZB0f94x88a8yYip49epSWtW8f/7q+/FSZ0LV6DBtW+ad77skb4XZH/aVDhy4/6/dpiPnz5Y1Nm
+j//E2uUhSi1auN1QsXWjabzfELE/RQm23dhp0xf7/rkuRk9am2bR0Xd+2qUffubi02tqoqNpaLevfm
grZtHYciIgyHiegHot5pqr6+QmbH3PmwG++aT378GGZnJMTkpiToxrKy0nVNFEjIog6dRJJSHBz796
VpRdcUG3q1q16b1CQ7NC06G1RUbFGfXvNiTlnxPbtYbPS09Urx0LScjJUYIqKhRVp0bcYmNF4uNdHB
9fZb7gAs3YrZtrT1CQusPhCNveunU3v7NdW1JSknuRw+F+NDqae2iaJiJ1Zw0HhYWqM2facmbNsg85m
XNs9mRBxy/t0tKGaOeff+r+AH0WRNcUjIjrmPHjqn6sobYtImvf0opZVtKir7kdxMmVGobNhT+FhER
cktMTI8f9eWnG3Pq7cxKuKoaCoiqsonEpChdShSlTb1+31NNZG+ESFgvovB4Ii3e7aZoTRM1KEh1Ggy
qkUh+I3IXKEriMX/T4VsK8/dRRG17Kkp4ggjHuVzV0UQqGQzkDAoKlhJxpxAFFahqb7+zSE8l5tTB1Z
Xu0YqiaKqql1lJy+0m1WxWQzRNXRcfHx/gQZDQJMwcPn487w50tzwRuSzz3IslZVpj+rrA8A54vnne
Wl9NzXvuafCYbGk7igqKvK5hgEAnOV+/VV61XdTMzHRLYcPZ+ba7Rk+HngLcPbBSlk+fPqPRBiN/pf0
V1Wi++83u+Pi3J9t3dr7v/pygLMRkoUPLpfCEuB22uTJldr06RVHnc6Il6+9VvFatBbgbIRk4Ye/ZNG
hA9H995srIyPVDW3adNunLwc4WyFZ+ODvuQ9ERLfcUu4aPty5y+1uf8Y9fhCgJSFZ+OB01jwAWG/QIL
csWFBmZFb+FhMT43NJd4CzFZKFH/qvIcHBRMnJZmfXrtpH0dHx2+qWApz9kCx8qKryXiN09uwqmjTJd
tjtjvyHopyeZ00CnE5IFj5UVipZzDSY6PeYPbt0cNu2VTOio7s02x03AQAAAAAAAAAAAAAAAAAAAA
AA
AA
AAADgFpt/tZ3jbeCAM
X0AAAAASUVORK5CYII=\"}\"

5 }

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 title: String,  
3     val content: String,  
4     val status: Int,  
5     val createdAt: Long,  
6     val updatedAt: Long,  
7     val id: Long,  
8     val userId: Long,  
9     val groupId: Long,  
10    val parentId: Long,  
11    val children: List<TodoTask>,  
12    val tags: List<String>,  
13    val isFavorite: Boolean,  
14    val isDeleted: Boolean,  
15    val isArchived: Boolean,  
16    val isShared: Boolean,  
17    val isPublic: Boolean,  
18    val isPrivate: Boolean,  
19    val isHidden: Boolean,  
20    val isVisible: Boolean,  
21    val isReadOnly: Boolean,  
22    val isEditable: Boolean,  
23    val isDeletable: Boolean,  
24    val isMoveable: Boolean,  
25    val isCopyable: Boolean,  
26    val isPasteable: Boolean,  
27    val isPrintable: Boolean,  
28    val isDownloadable: Boolean,  
29    val isUploadable: Boolean,  
30    val isShareable: Boolean,  
31    val isLinkable: Boolean,  
32    val isBookmarkable: Boolean,  
33    val isFavoritable: Boolean,  
34    val isCommentable: Boolean,  
35    val isRateable: Boolean,  
36    val isVoteable: Boolean,  
37    val isLikeable: Boolean,  
38    val isDislikeable: Boolean,  
39    val isStarable: Boolean,  
40    val isUnstarable: Boolean,  
41    val isBookmarkable: Boolean,  
42    val isFavoritable: Boolean,  
43    val isCommentable: Boolean,  
44    val isRateable: Boolean,  
45    val isVoteable: Boolean,  
46    val isLikeable: Boolean,  
47    val isDislikeable: Boolean,  
48    val isStarable: Boolean,  
49    val isUnstarable: Boolean,  
50    val isBookmarkable: Boolean,  
51    val isFavoritable: Boolean,  
52    val isCommentable: Boolean,  
53    val isRateable: Boolean,  
54    val isVoteable: Boolean,  
55    val isLikeable: Boolean,  
56    val isDislikeable: Boolean,  
57    val isStarable: Boolean,  
58    val isUnstarable: Boolean,  
59    val isBookmarkable: Boolean,  
60    val isFavoritable: Boolean,  
61    val isCommentable: Boolean,  
62    val isRateable: Boolean,  
63    val isVoteable: Boolean,  
64    val isLikeable: Boolean,  
65    val isDislikeable: Boolean,  
66    val isStarable: Boolean,  
67    val isUnstarable: Boolean,  
68    val isBookmarkable: Boolean,  
69    val isFavoritable: Boolean,  
70    val isCommentable: Boolean,  
71    val isRateable: Boolean,  
72    val isVoteable: Boolean,  
73    val isLikeable: Boolean,  
74    val isDislikeable: Boolean,  
75    val isStarable: Boolean,  
76    val isUnstarable: Boolean,  
77    val isBookmarkable: Boolean,  
78    val isFavoritable: Boolean,  
79    val isCommentable: Boolean,  
80    val isRateable: Boolean,  
81    val isVoteable: Boolean,  
82    val isLikeable: Boolean,  
83    val isDislikeable: Boolean,  
84    val isStarable: Boolean,  
85    val isUnstarable: Boolean,  
86    val isBookmarkable: Boolean,  
87    val isFavoritable: Boolean,  
88    val isCommentable: Boolean,  
89    val isRateable: Boolean,  
90    val isVoteable: Boolean,  
91    val isLikeable: Boolean,  
92    val isDislikeable: Boolean,  
93    val isStarable: Boolean,  
94    val isUnstarable: Boolean,  
95    val isBookmarkable: Boolean,  
96    val isFavoritable: Boolean,  
97    val isCommentable: Boolean,  
98    val isRateable: Boolean,  
99    val isVoteable: Boolean,  
100   val isLikeable: Boolean,  
101   val isDislikeable: Boolean,  
102   val isStarable: Boolean,  
103   val isUnstarable: Boolean,  
104   val isBookmarkable: Boolean,  
105   val isFavoritable: Boolean,  
106   val isCommentable: Boolean,  
107   val isRateable: Boolean,  
108   val isVoteable: Boolean,  
109   val isLikeable: Boolean,  
110   val isDislikeable: Boolean,  
111   val isStarable: Boolean,  
112   val isUnstarable: Boolean,  
113   val isBookmarkable: Boolean,  
114   val isFavoritable: Boolean,  
115   val isCommentable: Boolean,  
116   val isRateable: Boolean,  
117   val isVoteable: Boolean,  
118   val isLikeable: Boolean,  
119   val isDislikeable: Boolean,  
120   val isStarable: Boolean,  
121   val isUnstarable: Boolean,  
122   val isBookmarkable: Boolean,  
123   val isFavoritable: Boolean,  
124   val isCommentable: Boolean,  
125   val isRateable: Boolean,  
126   val isVoteable: Boolean,  
127   val isLikeable: Boolean,  
128   val isDislikeable: Boolean,  
129   val isStarable: Boolean,  
130   val isUnstarable: Boolean,  
131   val isBookmarkable: Boolean,  
132   val isFavoritable: Boolean,  
133   val isCommentable: Boolean,  
134   val isRateable: Boolean,  
135   val isVoteable: Boolean,  
136   val isLikeable: Boolean,  
137   val isDislikeable: Boolean,  
138   val isStarable: Boolean,  
139   val isUnstarable: Boolean,  
140   val isBookmarkable: Boolean,  
141   val isFavoritable: Boolean,  
142   val isCommentable: Boolean,  
143   val isRateable: Boolean,  
144   val isVoteable: Boolean,  
145   val isLikeable: Boolean,  
146   val isDislikeable: Boolean,  
147   val isStarable: Boolean,  
148   val isUnstarable: Boolean,  
149   val isBookmarkable: Boolean,  
150   val isFavoritable: Boolean,  
151   val isCommentable: Boolean,  
152   val isRateable: Boolean,  
153   val isVoteable: Boolean,  
154   val isLikeable: Boolean,  
155   val isDislikeable: Boolean,  
156   val isStarable: Boolean,  
157   val isUnstarable: Boolean,  
158   val isBookmarkable: Boolean,  
159   val isFavoritable: Boolean,  
160   val isCommentable: Boolean,  
161   val isRateable: Boolean,  
162   val isVoteable: Boolean,  
163   val isLikeable: Boolean,  
164   val isDislikeable: Boolean,  
165   val isStarable: Boolean,  
166   val isUnstarable: Boolean,  
167   val isBookmarkable: Boolean,  
168   val isFavoritable: Boolean,  
169   val isCommentable: Boolean,  
170   val isRateable: Boolean,  
171   val isVoteable: Boolean,  
172   val isLikeable: Boolean,  
173   val isDislikeable: Boolean,  
174   val isStarable: Boolean,  
175   val isUnstarable: Boolean,  
176   val isBookmarkable: Boolean,  
177   val isFavoritable: Boolean,  
178   val isCommentable: Boolean,  
179   val isRateable: Boolean,  
180   val isVoteable: Boolean,  
181   val isLikeable: Boolean,  
182   val isDislikeable: Boolean,  
183   val isStarable: Boolean,  
184   val isUnstarable: Boolean,  
185   val isBookmarkable: Boolean,  
186   val isFavoritable: Boolean,  
187   val isCommentable: Boolean,  
188   val isRateable: Boolean,  
189   val isVoteable: Boolean,  
190   val isLikeable: Boolean,  
191   val isDislikeable: Boolean,  
192   val isStarable: Boolean,  
193   val isUnstarable: Boolean,  
194   val isBookmarkable: Boolean,  
195   val isFavoritable: Boolean,  
196   val isCommentable: Boolean,  
197   val isRateable: Boolean,  
198   val isVoteable: Boolean,  
199   val isLikeable: Boolean,  
200   val isDislikeable: Boolean,  
201   val isStarable: Boolean,  
202   val isUnstarable: Boolean,  
203   val isBookmarkable: Boolean,  
204   val isFavoritable: Boolean,  
205   val isCommentable: Boolean,  
206   val isRateable: Boolean,  
207   val isVoteable: Boolean,  
208   val isLikeable: Boolean,  
209   val isDislikeable: Boolean,  
210   val isStarable: Boolean,  
211   val isUnstarable: Boolean,  
212   val isBookmarkable: Boolean,  
213   val isFavoritable: Boolean,  
214   val isCommentable: Boolean,  
215   val isRateable: Boolean,  
216   val isVoteable: Boolean,  
217   val isLikeable: Boolean,  
218   val isDislikeable: Boolean,  
219   val isStarable: Boolean,  
220   val isUnstarable: Boolean,  
221   val isBookmarkable: Boolean,  
222   val isFavoritable: Boolean,  
223   val isCommentable: Boolean,  
224   val isRateable: Boolean,  
225   val isVoteable: Boolean,  
226   val isLikeable: Boolean,  
227   val isDislikeable: Boolean,  
228   val isStarable: Boolean,  
229   val isUnstarable: Boolean,  
230   val isBookmarkable: Boolean,  
231   val isFavoritable: Boolean,  
232   val isCommentable: Boolean,  
233   val isRateable: Boolean,  
234   val isVoteable: Boolean,  
235   val isLikeable: Boolean,  
236   val isDislikeable: Boolean,  
237   val isStarable: Boolean,  
238   val isUnstarable: Boolean,  
239   val isBookmarkable: Boolean,  
240   val isFavoritable: Boolean,  
241   val isCommentable: Boolean,  
242   val isRateable: Boolean,  
243   val isVoteable: Boolean,  
244   val isLikeable: Boolean,  
245   val isDislikeable: Boolean,  
246   val isStarable: Boolean,  
247   val isUnstarable: Boolean,  
248   val isBookmarkable: Boolean,  
249   val isFavoritable: Boolean,  
250   val isCommentable: Boolean,  
251   val isRateable: Boolean,  
252   val isVoteable: Boolean,  
253   val isLikeable: Boolean,  
254   val isDislikeable: Boolean,  
255   val isStarable: Boolean,  
256   val isUnstarable: Boolean,  
257   val isBookmarkable: Boolean,  
258   val isFavoritable: Boolean,  
259   val isCommentable: Boolean,  
260   val isRateable: Boolean,  
261   val isVoteable: Boolean,  
262   val isLikeable: Boolean,  
263   val isDislikeable: Boolean,  
264   val isStarable: Boolean,  
265   val isUnstarable: Boolean,  
266   val isBookmarkable: Boolean,  
267   val isFavoritable: Boolean,  
268   val isCommentable: Boolean,  
269   val isRateable: Boolean,  
270   val isVoteable: Boolean,  
271   val isLikeable: Boolean,  
272   val isDislikeable: Boolean,  
273   val isStarable: Boolean,  
274   val isUnstarable: Boolean,  
275   val isBookmarkable: Boolean,  
276   val isFavoritable: Boolean,  
277   val isCommentable: Boolean,  
278   val isRateable: Boolean,  
279   val isVoteable: Boolean,  
280   val isLikeable: Boolean,  
281   val isDislikeable: Boolean,  
282   val isStarable: Boolean,  
283   val isUnstarable: Boolean,  
284   val isBookmarkable: Boolean,  
285   val isFavoritable: Boolean,  
286   val isCommentable: Boolean,  
287   val isRateable: Boolean,  
288   val isVoteable: Boolean,  
289   val isLikeable: Boolean,  
290   val isDislikeable: Boolean,  
291   val isStarable: Boolean,  
292   val isUnstarable: Boolean,  
293   val isBookmarkable: Boolean,  
294   val isFavoritable: Boolean,  
295   val isCommentable: Boolean,  
296   val isRateable: Boolean,  
297   val isVoteable: Boolean,  
298   val isLikeable: Boolean,  
299   val isDislikeable: Boolean,  
300   val isStarable: Boolean,  
301   val isUnstarable: Boolean,  
302   val isBookmarkable: Boolean,  
303   val isFavoritable: Boolean,  
304   val isCommentable: Boolean,  
305   val isRateable: Boolean,  
306   val isVoteable: Boolean,  
307   val isLikeable: Boolean,  
308   val isDislikeable: Boolean,  
309   val isStarable: Boolean,  
310   val isUnstarable: Boolean,  
311   val isBookmarkable: Boolean,  
312   val isFavoritable: Boolean,  
313   val isCommentable: Boolean,  
314   val isRateable: Boolean,  
315   val isVoteable: Boolean,  
316   val isLikeable: Boolean,  
317   val isDislikeable: Boolean,  
318   val isStarable: Boolean,  
319   val isUnstarable: Boolean,  
320   val isBookmarkable: Boolean,  
321   val isFavoritable: Boolean,  
322   val isCommentable: Boolean,  
323   val isRateable: Boolean,  
324   val isVoteable: Boolean,  
325   val isLikeable: Boolean,  
326   val isDislikeable: Boolean,  
327   val isStarable: Boolean,  
328   val isUnstarable: Boolean,  
329   val isBookmarkable: Boolean,  
330   val isFavoritable: Boolean,  
331   val isCommentable: Boolean,  
332   val isRateable: Boolean,  
333   val isVoteable: Boolean,  
334   val isLikeable: Boolean,  
335   val isDislikeable: Boolean,  
336   val isStarable: Boolean,  
337   val isUnstarable: Boolean,  
338   val isBookmarkable: Boolean,  
339   val isFavoritable: Boolean,  
340   val isCommentable: Boolean,  
341   val isRateable: Boolean,  
342   val isVoteable: Boolean,  
343   val isLikeable: Boolean,  
344   val isDislikeable: Boolean,  
345   val isStarable: Boolean,  
346   val isUnstarable: Boolean,  
347   val isBookmarkable: Boolean,  
348   val isFavoritable: Boolean,  
349   val isCommentable: Boolean,  
350   val isRateable: Boolean,  
351   val isVoteable: Boolean,  
352   val isLikeable: Boolean,  
353   val isDislikeable: Boolean,  
354   val isStarable: Boolean,  
355   val isUnstarable: Boolean,  
356   val isBookmarkable: Boolean,  
357   val isFavoritable: Boolean,  
358   val isCommentable: Boolean,  
359   val isRateable: Boolean,  
360   val isVoteable: Boolean,  
361   val isLikeable: Boolean,  
362   val isDislikeable: Boolean,  
363   val isStarable: Boolean,  
364   val isUnstarable: Boolean,  
365   val isBookmarkable: Boolean,  
366   val isFavoritable: Boolean,  
367   val isCommentable: Boolean,  
368   val isRateable: Boolean,  
369   val isVoteable: Boolean,  
370   val isLikeable: Boolean,  
371   val isDislikeable: Boolean,  
372   val isStarable: Boolean,  
373   val isUnstarable: Boolean,  
374   val isBookmarkable: Boolean,  
375   val isFavoritable: Boolean,  
376   val isCommentable: Boolean,  
377   val isRateable: Boolean,  
378   val isVoteable: Boolean,  
379   val isLikeable: Boolean,  
380   val isDislikeable: Boolean,  
381   val isStarable: Boolean,  
382   val isUnstarable: Boolean,  
383   val isBookmarkable: Boolean,  
384   val isFavoritable: Boolean,  
385   val isCommentable: Boolean,  
386   val isRateable: Boolean,  
387   val isVoteable: Boolean,  
388   val isLikeable: Boolean,  
389   val isDislikeable: Boolean,  
390   val isStarable: Boolean,  
391   val isUnstarable: Boolean,  
392   val isBookmarkable: Boolean,  
393   val isFavoritable: Boolean,  
394   val isCommentable: Boolean,  
395   val isRateable: Boolean,  
396   val isVoteable: Boolean,  
397   val isLikeable: Boolean,  
398   val isDislikeable: Boolean,  
399   val isStarable: Boolean,  
400   val isUnstarable: Boolean,  
401   val isBookmarkable: Boolean,  
402   val isFavoritable: Boolean,  
403   val isCommentable: Boolean,  
404   val isRateable: Boolean,  
405   val isVoteable: Boolean,  
406   val isLikeable: Boolean,  
407   val isDislikeable: Boolean,  
408   val isStarable: Boolean,  
409   val isUnstarable: Boolean,  
410   val isBookmarkable: Boolean,  
411   val isFavoritable: Boolean,  
412   val isCommentable: Boolean,  
413   val isRateable: Boolean,  
414   val isVoteable: Boolean,  
415   val isLikeable: Boolean,  
416   val isDislikeable: Boolean,  
417   val isStarable: Boolean,  
418   val isUnstarable: Boolean,  
419   val isBookmarkable: Boolean,  
420   val isFavoritable: Boolean,  
421   val isCommentable: Boolean,  
422   val isRateable: Boolean,  
423   val isVoteable: Boolean,  
424   val isLikeable: Boolean,  
425   val isDislikeable: Boolean,  
426   val isStarable: Boolean,  
427   val isUnstarable: Boolean,  
428   val isBookmarkable: Boolean,  
429   val isFavoritable: Boolean,  
430   val isCommentable: Boolean,  
431   val isRateable: Boolean,  
432   val isVoteable: Boolean,  
433   val isLikeable: Boolean,  
434   val isDislikeable: Boolean,  
435   val isStarable: Boolean,  
436   val isUnstarable: Boolean,  
437   val isBookmarkable: Boolean,  
438   val isFavoritable: Boolean,  
439   val isCommentable: Boolean,  
440   val isRateable: Boolean,  
441   val isVoteable: Boolean,  
442   val isLikeable: Boolean,  
443   val isDislikeable: Boolean,  
444   val isStarable: Boolean,  
445   val isUnstarable: Boolean,  
446   val isBookmarkable: Boolean,  
447   val isFavoritable: Boolean,  
448   val isCommentable: Boolean,  
449   val isRateable: Boolean,  
450   val isVoteable: Boolean,  
451   val isLikeable: Boolean,  
452   val isDislikeable: Boolean,  
453   val isStarable: Boolean,  
454   val isUnstarable: Boolean,  
455   val isBookmarkable: Boolean,  
456   val isFavoritable: Boolean,  
457   val isCommentable: Boolean,  
458   val isRateable: Boolean,  
459   val isVoteable: Boolean,  
460   val isLikeable: Boolean,  
461   val isDislikeable: Boolean,  
462   val isStarable: Boolean,  
463   val isUnstarable: Boolean,  
464   val isBookmarkable: Boolean,  
465   val isFavoritable: Boolean,  
466   val isCommentable: Boolean,  
467   val isRateable: Boolean,  
468   val isVoteable: Boolean,  
469   val isLikeable: Boolean,  
470   val isDislikeable: Boolean,  
471   val isStarable: Boolean,  
472   val isUnstarable: Boolean,  
473   val isBookmarkable: Boolean,  
474   val isFavoritable: Boolean,  
475   val isCommentable: Boolean,  
476   val isRateable: Boolean,  
477   val isVoteable: Boolean,  
478   val isLikeable: Boolean,  
479   val isDislikeable: Boolean,  
480   val isStarable: Boolean,  
481   val isUnstarable: Boolean,  
482   val isBookmarkable: Boolean,  
483   val isFavoritable: Boolean,  
484   val isCommentable: Boolean,  
485   val isRateable: Boolean,  
486   val isVoteable: Boolean,  
487   val isLikeable: Boolean,  
488   val isDislikeable: Boolean,  
489   val isStarable: Boolean,  
490   val isUnstarable: Boolean,  
491   val isBookmarkable: Boolean,  
492   val isFavoritable: Boolean,  
493   val isCommentable: Boolean,  
494   val isRateable: Boolean,  
495   val isVoteable: Boolean,  
496   val isLikeable: Boolean,  
497   val isDislikeable: Boolean,  
498   val isStarable: Boolean,  
499   val isUnstarable: Boolean,  
500   val isBookmarkable: Boolean,  
501   val isFavoritable: Boolean,  
502   val isCommentable: Boolean,  
503   val isRateable: Boolean,  
504   val isVoteable: Boolean,  
505   val isLikeable: Boolean,  
506   val isDislikeable: Boolean,  
507   val isStarable: Boolean,  
508   val isUnstarable: Boolean,  
509   val isBookmarkable: Boolean,  
510   val isFavoritable: Boolean,  
511   val isCommentable: Boolean,  
512   val isRateable: Boolean,  
513   val isVoteable: Boolean,  
514   val isLikeable: Boolean,  
515   val isDislikeable: Boolean,  
516   val isStarable: Boolean,  
517   val isUnstarable: Boolean,  
518   val isBookmarkable: Boolean,  
519   val isFavoritable: Boolean,  
520   val isCommentable: Boolean,  
521   val isRateable: Boolean,  
522   val isVoteable: Boolean,  
523   val isLikeable: Boolean,  
524   val isDislikeable: Boolean,  
525   val isStarable: Boolean,  
526   val isUnstarable: Boolean,  
527   val isBookmarkable: Boolean,  
528   val isFavoritable: Boolean,  
529   val isCommentable: Boolean,  
530   val isRateable: Boolean,  
531   val isVoteable: Boolean,  
532   val isLikeable: Boolean,  
533   val isDislikeable: Boolean,  
534   val isStarable: Boolean,  
535   val isUnstarable: Boolean,  
536   val isBookmarkable: Boolean,  
537   val isFavoritable: Boolean,  
538   val isCommentable: Boolean,  
539   val isRateable: Boolean,  
540   val isVoteable: Boolean,  
541   val isLikeable: Boolean,  
542   val isDislikeable: Boolean,  
543   val isStarable: Boolean,  
544   val isUnstarable: Boolean,  
545   val isBookmarkable: Boolean,  
546   val isFavoritable: Boolean,  
547   val isCommentable: Boolean,  
548   val isRateable: Boolean,  
549   val isVoteable: Boolean,  
550   val isLikeable: Boolean,  
551   val isDislikeable: Boolean,  
552   val isStarable: Boolean,  
553   val isUnstarable: Boolean,  
554   val isBookmarkable: Boolean,  
555   val isFavoritable: Boolean,  
556   val isCommentable: Boolean,  
557   val isRateable: Boolean,  
558   val isVoteable: Boolean,  
559   val isLikeable: Boolean,  
560   val isDislikeable: Boolean,  
561   val isStarable: Boolean,  
562   val isUnstarable: Boolean,  
563   val isBookmarkable: Boolean,  
564   val isFavoritable: Boolean,  
565   val isCommentable: Boolean,  
566   val isRateable: Boolean,  
567   val isVoteable: Boolean,  
568   val isLikeable: Boolean,  
569   val isDislikeable: Boolean,  
570   val isStarable: Boolean,  
571   val isUnstarable: Boolean,  
572   val isBookmarkable: Boolean,  
573   val isFavoritable: Boolean,  
574   val isCommentable: Boolean,  
575   val isRateable: Boolean,  
576   val isVoteable: Boolean,  
577   val isLikeable: Boolean,  
578   val isDislikeable: Boolean,  
579   val isStarable: Boolean,  
580   val isUnstarable: Boolean,  
581   val isBookmarkable: Boolean,  
582   val isFavoritable: Boolean,  
583   val isCommentable: Boolean,  
584   val isRateable: Boolean,  
585   val isVoteable: Boolean,  
586   val isLikeable: Boolean,  
587   val isDislikeable: Boolean,  
588   val isStarable: Boolean,  
589   val isUnstarable: Boolean,  
590   val isBookmarkable: Boolean,  
591   val isFavoritable: Boolean,  
592   val isCommentable: Boolean,  
593   val isRateable: Boolean,  
594   val isVoteable: Boolean,  
595   val isLikeable: Boolean,  
596   val isDislikeable: Boolean,  
597   val isStarable: Boolean,  
598   val isUnstarable: Boolean,  
599   val isBookmarkable: Boolean,  
600   val isFavoritable: Boolean,  
601   val isCommentable: Boolean,  
602   val isRateable: Boolean,  
603   val isVoteable: Boolean,  
604   val isLikeable: Boolean,  
605   val isDislikeable: Boolean,  
606   val isStarable: Boolean,  
607   val isUnstarable: Boolean,  
608   val isBookmarkable: Boolean,  
609   val isFavoritable: Boolean,  
610   val isCommentable: Boolean,  
611   val isRateable: Boolean,  
612   val isVoteable: Boolean,  
613   val isLikeable: Boolean,  
614   val isDislikeable: Boolean,  
615   val isStarable: Boolean,  
616   val isUnstarable: Boolean,  
617   val isBookmarkable: Boolean,  
618   val isFavoritable: Boolean,  
619   val isCommentable: Boolean,  
620   val isRateable: Boolean,  
621   val isVoteable: Boolean,  
622   val isLikeable: Boolean,  
623   val isDislikeable: Boolean,  
624   val isStarable: Boolean,  
625   val isUnstarable: Boolean,  
626   val isBookmarkable: Boolean,  
627   val isFavoritable: Boolean,  
628   val isCommentable: Boolean,  
629   val isRateable: Boolean,  
630   val isVoteable: Boolean,  
631   val isLikeable: Boolean,  
632   val isDislikeable: Boolean,  
633   val isStarable: Boolean,  
634   val isUnstarable: Boolean,  
635   val isBookmarkable: Boolean,  
636   val isFavoritable: Boolean,  
637   val isCommentable: Boolean,  
638   val isRateable: Boolean,  
639   val isVoteable: Boolean,  
640   val isLikeable: Boolean,  
641   val isDislikeable: Boolean,  
642   val isStarable: Boolean,  
643   val isUnstarable: Boolean,  
644   val isBookmarkable: Boolean,  
645   val isFavoritable: Boolean,  
646   val isCommentable: Boolean,  
647   val isRateable: Boolean,  
648   val isVoteable: Boolean,  
649   val isLikeable: Boolean,  
650   val isDislikeable: Boolean,  
651   val isStarable: Boolean,  
652   val isUnstarable: Boolean,  
653   val isBookmarkable: Boolean,  
654   val isFavoritable: Boolean,  
655   val isCommentable: Boolean,  
656   val isRateable: Boolean,  
657   val isVoteable: Boolean,  
658   val isLikeable: Boolean,  
659   val isDislikeable: Boolean,  
660   val isStarable: Boolean,  
661   val isUnstarable: Boolean,  
662   val isBookmarkable: Boolean,  
663   val isFavoritable: Boolean,  
664   val isCommentable: Boolean,  
665   val isRateable: Boolean,  
666   val isVoteable: Boolean,  
667   val isLikeable: Boolean,  
668   val isDislikeable: Boolean,  
669   val isStarable: Boolean,  
670   val isUnstarable: Boolean,  
671   val isBookmarkable: Boolean,  
672   val isFavoritable: Boolean,  
673   val isCommentable: Boolean,  
674   val isRateable: Boolean,  
675   val isVoteable: Boolean,  
676   val isLikeable: Boolean,  
677   val isDislikeable: Boolean,  
678   val isStarable: Boolean,  
679   val isUnstarable: Boolean,  
680   val isBookmarkable: Boolean,  
681   val isFavoritable: Boolean,  
682   val isCommentable: Boolean,  
683   val isRateable: Boolean,  
684   val isVoteable: Boolean,  
685   val isLikeable: Boolean,  
686   val isDislikeable: Boolean,  
687   val isStarable: Boolean,  
688   val isUnstarable: Boolean,  
689   val isBookmarkable: Boolean,  
690   val isFavoritable: Boolean,  
691   val isCommentable: Boolean,  
692   val isRateable: Boolean,  
693   val isVoteable: Boolean,  
694   val isLikeable: Boolean,  
695   val isDislikeable: Boolean,  
696   val isStarable: Boolean,  
697   val isUnstarable: Boolean,  
698   val isBookmarkable: Boolean,  
699   val isFavoritable: Boolean,  
700   val isCommentable: Boolean,  
701   val isRateable: Boolean,  
702   val isVoteable: Boolean,  
703   val isLikeable: Boolean,  
704   val isDislikeable: Boolean,  
705   val isStarable: Boolean,  
706   val isUnstarable: Boolean,  
707   val isBookmarkable: Boolean,  
708   val isFavoritable: Boolean,  
709   val isCommentable: Boolean,  
710   val isRateable: Boolean,  
711   val isVoteable: Boolean,  
712   val isLikeable: Boolean,  
713   val isDislikeable: Boolean,  
714   val isStarable: Boolean,  
715   val isUnstarable: Boolean,  
716   val isBookmarkable: Boolean,  
717   val isFavoritable: Boolean,  
718   val isCommentable: Boolean,  
719   val isRateable: Boolean,  
720   val isVoteable: Boolean,  
721   val isLikeable: Boolean,  
722   val isDislikeable: Boolean,  
723   val isStarable: Boolean,  
724   val isUnstarable: Boolean,  
725   val isBookmarkable: Boolean,  
726   val isFavoritable: Boolean,  
727   val isCommentable: Boolean,  
728   val isRateable: Boolean,  
729   val isVoteable: Boolean,  
730   val isLikeable: Boolean,  
731   val isDislikeable: Boolean,  
732   val isStarable: Boolean,  
733   val isUnstarable: Boolean,  
734   val isBookmarkable: Boolean,  
735   val isFavoritable: Boolean,  
736   val isCommentable: Boolean,  
737   val isRateable: Boolean,  
738   val isVoteable: Boolean,  
739   val isLikeable: Boolean,  
740   val isDislikeable: Boolean,  
741   val isStarable: Boolean,  
742   val isUnstarable: Boolean,  
743   val isBookmarkable: Boolean,  
744   val isFavoritable: Boolean,  
745   val isCommentable: Boolean,  
746   val isRateable: Boolean,  
747   val isVoteable: Boolean,  
748   val isLikeable: Boolean,  
749   val isDislikeable: Boolean,  
750   val isStarable: Boolean,  
751   val isUnstarable: Boolean,  
752   val isBookmarkable: Boolean,  
753   val isFavoritable: Boolean,  
754   val isCommentable: Boolean,  
755   val isRateable: Boolean,  
756   val isVoteable: Boolean,  
757   val isLikeable: Boolean,  
758   val isDislikeable: Boolean,  
759   val isStarable: Boolean,  
760   val isUnstarable: Boolean,  
761   val isBookmarkable: Boolean,  
762   val isFavoritable: Boolean,  
763   val isCommentable: Boolean,  
764   val isRateable: Boolean,  
765   val isVoteable: Boolean,  
766   val isLikeable: Boolean,  
767   val isDislikeable: Boolean,  
768   val isStarable: Boolean,  
769   val isUnstarable: Boolean,  
770   val isBookmarkable: Boolean,  
771   val isFavoritable: Boolean,  
772   val isCommentable: Boolean,  
773   val isRateable: Boolean,  
774   val isVoteable: Boolean,  
775   val isLikeable: Boolean,  
776   val isDislikeable: Boolean,  
777   val isStarable: Boolean,  
778   val isUnstarable: Boolean,  
779   val isBookmarkable: Boolean,  
780   val isFavoritable: Boolean,  
781   val isCommentable: Boolean,  
782   val isRateable: Boolean,  
783   val isVoteable: Boolean,  
784   val isLikeable: Boolean,  
785   val isDislikeable: Boolean,  
786   val isStarable: Boolean,  
787   val isUnstarable: Boolean,  
788   val isBookmarkable: Boolean,  
789   val isFavoritable: Boolean,  
790   val isCommentable: Boolean,  
791   val isRateable: Boolean,  
792   val isVoteable: Boolean,  
793   val isLikeable: Boolean,  
794   val isDislikeable: Boolean,  
795   val isStarable: Boolean,  
796   val isUnstarable: Boolean,  
79
```



```

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         "createTime": "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 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpzdmU6MDF9.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 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpzdmU6MDF9.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         "createTime": "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 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpzdmU6ImF9.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/avataaars/svg?seed=Zhang"
9          },
10         {
11             "userId": 1002,
12             "username": "LiSi",
13             "avatarUrl": "https://api.dicebear.com/7.x/avataaars/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 }
```