

聊天消息协议

聊天消息协议

WebSocket 路径

JSON 格式

字段说明

msgId 消息编号

conversationId 会话编号

type 消息类型

from 发送人

sendTo 消息接收人

read 消息已读标志

status 消息状态

time 消息发送时间

客户端消息样例

发送消息

收到消息

客服端消息样例

有客户分配过来建立会话

收到客户发送过来的消息

发送文本消息到会话中

发送图片消息到会话中

接收到图片消息

发送输入中提示

发送消息已读反馈

WebSocket 路径

客户用户端: `ws://localhost:9001/online/api/v1/websocket/customer/用户的token`

客服端: `ws://localhost:9001/online/api/v1/websocket/seatUser/客服登陆后获取到的token`

JSON 格式

消息协议格式如下:

```
{  
  "content": "消息内容,格式根据type会有变动. 普通消息就是string, 附件消息会有文件  
大小, 文件类型等字段描述",
```

```
    "conversationId": "conversationId, 客服端需要填充, 指定将消息发送到哪个会话中, 客户端无需填写",
    "from": "sendFrom, 后端自动填充",
    "msgId": "3438e05318d64bdbb196c51e53f3d5a0, 消息编号, 前端生成. 内容为UUID即可, 需要保证消息编号不重复.",
    "read": "0, 消息是否已读(0未读1已读), 后端填充. 拉取历史消息时会有此字段.",
    "sendTo": [
      {
        "userId": "1, 用户编号. sendTo字段由后端自动填充"
      }
    ],
    "senderType": "seatUser, 发送人的用户类型, 后端填充",
    "status": "normal, 消息状态类型(normal正常, del删除, recall撤回)",
    "time": 1562034342581,
    "type": "message, 消息类型, 详见后面解释"
  }
}
```

字段说明

msgId 消息编号

`msgId` 为消息编号, 格式为 UUID, 由 前端/发送方 生成并发送给后端, 需要保证生成的 id 唯一

conversationId 会话编号

`conversationId` 为会话编号, 将每个客户与客服的聊天抽象为一个会话(conversation);

每个会话有且仅有一个客户, 可以有多个客服, 此状态由后端维护;

客户端发送消息不需要传该参数, 后端自动识别并给消息加上编号

客服端发送消息需要添加会话编号, 用于给后端区分要发送到的会话.

type 消息类型

字段类型为 *string*, 目前支持的消息类型有:

- `message` 文本消息
 - 对应的消息 `content` 字段填写消息内容即可.
- `image` 图片消息
 - 图片类型的消息.
- `heartBeat` 心跳消息.

- 客户端和客服端都可以收到, 由客户端主动发起, 服务端收到此消息后会回复一个心跳包给客户端.
- **readFeedback** 消息已读反馈
 - 前端收到一个聊天类型的消息并确保显示给了用户后发送给后端(此时消息content填写对应的msgId即可). 后端将此类型的消息转发给会话中其他人员.
- **finishService** 结束服务.
 - 客服和客户端都可以发送此消息, 表示要主动结束服务.
- **typing** 输入中提示.
 - 前端可以每5秒判断输入框中的内容不为空并且跟上次检测的时候内容不一样, 就可以发送typing类型的消息给后端.表示用户正在输入中. 另一端用户收到此类型的消息可以在聊天的状态提示框中提示对方输入中.
- **fail** 消息发送失败.
 - 系统级别的失败. 可能因为后端mq等宕机引起.
- **newConversation** 新会话提示
 - 用于提醒客服有新客户接入.在系统自动给客户分配一个客服人员后将消息推送给客服.
- **notification** 通知类消息, 由服务器发送, 如客服离线, 系统的一些提示, 可以考虑使用 toast 的形式.

from 发送人

from (:string) 字段为发送人编号

sendTo 消息接收人

字段类型为 *string*, 消息接收人编号

read 消息已读标志

字段类型 *string* 消息是否已读(0:未读,1:已读)

status 消息状态

字段类型 *string*, 标识消息目前的状态. 客户端拉取消息会提示

目前支持的消息状态有如下:

- **normal** 0, 正常状态, 默认值
 - **del** 1, 消息已被删除
 - **recall** 2, 消息被撤回
-

time 消息发送时间

字段类型: *timestamp* 消息到达服务器后, 由服务器进行填充.

客户端消息样例

发送消息

```
{
  "content": "你好,我想咨询一个事情",
  "msgId": "3438e05318d64bdbb196c51e53f3d5a0",
  "type": "message"
}
```

收到消息

```
{
  "read": "0",
  "from": "1",
  "senderType": "seatUser",
  "time": 1562858075132,
  "type": "message",
  "content": "系统已自动为您分配了客服, 客服编号:1",
  "status": "normal"
}
```

客服端消息样例

有客户分配过来建立会话

收到消息

```
{
  "read": "0",
  "sendTo": [
    {
      "userType": "seatUser",
      "userId": "1"
    }
  ]
}
```

```
],
  "conversationId": "7a0a550430d44379af0ca4207adedac2",
  "senderType": "customer",
  "time": 1562856453789,
  "type": "newConversation",
  "status": "normal"
}
```

收到客户发送过来的消息

```
{
  "read": "0",
  "sendTo": [
    {
      "userType": "seatUser",
      "userId": "1"
    }
  ],
  "conversationId": "7a0a550430d44379af0ca4207adedac2",
  "msgId": "jsklldgjlsgjlsdjgldsjsj",
  "from": "4a1ca803b2f448ed8eccea070e6886c2",
  "senderType": "customer",
  "time": 1562856841692,
  "type": "message",
  "content": "normal",
  "status": "normal"
}
```

发送文本消息到会话中

```
{
  "content": "你好，请问要咨询什么事情?",
  "conversationId": "7a0a550430d44379af0ca4207adedac2",
  "msgId": "1034bdb1eb1896c538e05353f3d5ad64",
  "type": "message"
}
```

发送图片消息到会话中

其中 content 字段为上传图片消息时返回的一些字段.

```
{
```

```
"conversationId":"7a0a550430d44379af0ca4207adedac2",
"msgId":"1034bdb1eb1896c538e05353f3d5ad64",
"type":"image",
"content":{
  "imgUrl":"http://localhost:9001/online//api/v1/chat/attachment/image/view/c7a8113dcfce48a7898ad2f2d00ccd7d.png",
  "imgName":"Screen Shot 2019-07-13 at 4.49.38 PM.png",
  "imgId":"c7a8113dcfce48a7898ad2f2d00ccd7d",
  "imgSize":1580801,
  "imgType":"png"
}
```

接收到图片消息

与接收普通文本消息一样,不同的是 `content` 字段与发送时的一致

发送输入中提示

```
{
  "content": "",
  "conversationId": "7a0a550430d44379af0ca4207adedac2",
  "msgId": "",
  "type": "typing"
}
```

发送消息已读反馈

`msgId` 字段为对应消息的编号,非随机生成.

```
{
  "content": "",
  "conversationId": "7a0a550430d44379af0ca4207adedac2",
  "msgId": "1034bdb1eb1896c538e05353f3d5ad64",
  "type": "readFeedback"
}
```