**前言**

API的发送方式均为socket

文档将分为两部分，第一部分为从client端发送到server端的API，第二部分为从server端发送到client端的API，发送时均打包成JSON格式

现制定API基本格式如下：

client -> server API:

|  |  |  |
| --- | --- | --- |
| 名称 | 是否必须 | 说明 |
| type | 是 | API类型 |
| data | 是 | API数据 |

server -> client API：

|  |  |  |
| --- | --- | --- |
| 名称 | 是否必须 | 说明 |
| type | 是 | API类型 |
| data | 是 | API数据 |
| msg | 否 | API附带的消息（如提示或者警告信息） |

需注意的三个基本点：

1. 什么时候需要发送该API
2. 发送给哪些socket对象，即发送给哪些用户
3. 什么时候需要去准备接受该API（网页端更需注意这点）

目录

[1. 客户端发送的API 3](#_Toc466470576)

[1.1 用户登录（type: login） 3](#_Toc466470577)

[1.2 加入房间（type: join\_room） 3](#_Toc466470578)

[1.3 文字消息（type: text\_message） 3](#_Toc466470579)

[2. 服务端发送的API 4](#_Toc466470580)

[2.1 用户登录成功（type: login\_succ） 4](#_Toc466470581)

[2.2 文字消息（type: text\_message） 4](#_Toc466470582)

# 客户端发送的API

#### 用户登录（type: login）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| name | 是 | string | 用户昵称 |

#### 加入房间（type: join\_room）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| number | 是 | int | 房间号，1000到9999之间的数 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

#### 文字消息（type: text\_message）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| message | 是 | string | 聊天文字消息 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

用户向房间的所有人发送文字消息

# 服务端发送的API

#### 用户登录成功（type: login\_succ）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| name | 是 | string | 用户昵称 |
| id | 是 | int | 用户id |
| token | 是 | string | 用户token |

用户登录后需返回改API

#### 文字消息（type: text\_message）

data:

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 是否必须 | 类型 | 说明 |
| message | 是 | string | 聊天文字消息 |
| id | 是 | int | 用户id，标记消息是谁发送的 |
| name | 是 | string | 消息发送人的昵称 |

服务端用来分发聊天信息给用户