|  |  |  |  |
| --- | --- | --- | --- |
| 文档状态 | [ √ ] 草稿 [ ] 已发布[ ] 修改中 | | |
| 文档编号 |  | | |
| 编 撰 |  | 编撰日期 | 2020-11-20 |
| 保密级别 | 非密 | 文档版本 | V1.0.0 |
| 部 门/组 | 软件产品事业部 | | |

**Socket协议文档**

|  |  |
| --- | --- |
| 拟 制： | sux |
| 审 核： |  |
| 批 准： |  |

**XXXX有限公司**

**二〇二〇年十一月**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **次序** | **版本** | **修订人** | **修订日期** | **修订摘要** |
| 1 | V1.0.0 | sux | 2020-11-20 | 创建文档 |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| 6 |  |  |  |  |
| 7 |  |  |  |  |
| 8 |  |  |  |  |
| 9 |  |  |  |  |
| 10 |  |  |  |  |

**目录**

[1 引言 3](#_Toc56761349)

[1.1 目的 3](#_Toc56761350)

[1.2 目标范围 3](#_Toc56761351)

[1.3 读者对象 3](#_Toc56761352)

[1.4 参考文献 3](#_Toc56761353)

[2 Socket消息通信协议 3](#_Toc56761354)

[2.1 Socket客户端连接注册心跳流程 3](#_Toc56761356)

[2.2 Socket客户端接收消息数据流程 4](#_Toc56761357)

[2.3 Socket客户端向服务端发送消息流程 4](#_Toc56761358)

[2.4 Socket数据包 4](#_Toc56761359)

[2.5 数据类型 5](#_Toc56761360)

[2.3.1 SocketRegisterData 客户端注册数据 5](#_Toc56761366)

[2.3.2 MsgContent 消息数据 5](#_Toc56761367)

[2.3.3 SocketResult 消息返回值数据 6](#_Toc56761368)

# 引言

## 目的

方便Socket通信对接

## 目标范围

定义了Socket通信流程、协议、数据类型

## 读者对象

开发人员。

## 参考文献

无

# Socket消息通信协议



## Socket客户端连接注册心跳流程

1.连接服务端

2.注册，客户端把自己的ID注册到服务端，ID可以是设备的唯一标识

3.维持心跳，建议10秒发一次心跳，服务端收到心跳包会立即发送心跳反馈包

4.客户端需要实现断线重连，如果15秒收不到心跳反馈包，可以认为服务端已离线，重新连接并注册

## Socket客户端接收消息数据流程

1.服务端向客户端发消息数据，数据中有CallbackId，业务数据序列化成Json字符串，赋给Content字段，Content字段是string类型，业务数据类型再另行约定。

2.客户端收到消息，立即发一条消息反馈包给服务端，消息反馈包中的CallbackId即是收到的消息数据中的CallbackId

## Socket客户端向服务端发送消息流程

1.客户端向服务端发送消息数据，数据中包含CallbackId，业务数据序列化成Json字符串，赋给Content字段，Content字段是string类型，业务数据类型再另行约定。

2.服务端收到消息，立即发一条消息反馈包给客户端，消息反馈包中的CallbackId即是服务端收到的消息数据中的CallbackId，客户端以此判断服务端已收到消息

## Socket数据包

|  |  |  |  |
| --- | --- | --- | --- |
| **字段** | **字段说明** | **字节长度** | **说明** |
| header | 消息头 | 4 | 字符串0XFF以ASCII编码转换成字节数组 |
| type | 消息类型 | 1 | 1：心跳  2：心跳应答  3：注册  4：注册反馈  5：消息数据  6：返回值 |
| length | 消息体长度 | 4 | 消息体长度 |
| body | 消息体 | 变长 | type=1、2、4没有length和body  type=3，body由SocketRegisterData序列化成Json字符串，再以UTF8编码转换成字节数组  type=5，body由MsgContent序列化成Json字符串，再以UTF8编码转换成字节数组  type=6，body由SocketResult序列化成Json字符串，再以UTF8编码转换成字节数组 |

## 数据类型



### SocketRegisterData 客户端注册数据

|  |  |  |
| --- | --- | --- |
| **SocketRegisterData** | | |
| **字段名称** | **字段类型** | **说明** |
| SocketClientId | string | Socket客户端ID |

### MsgContent 消息数据

|  |  |  |
| --- | --- | --- |
| **MsgContent** | | |
| **字段名称** | **字段类型** | **说明** |
| CallbackId | string | 回调ID |
| Content | string | 消息数据 |

### SocketResult 消息返回值数据

|  |  |  |
| --- | --- | --- |
| **SocketResult** | | |
| **字段名称** | **字段类型** | **说明** |
| success | bool | 是否成功 |
| errorMsg | string | 失败原因 |
| callbackId | string | 回调ID |