TCP长链接

1.概述

tcp长链接，在用户登录态下与服务器保持长链接，用户部分api请求和通知推送。

2.协议

参考http://wiki.cheyaoshicorp.com/pages/viewpage.action?pageId=5735102

头部

字节长度	备注
8	8字节前缀(包长长度)
2	协议版本
4	协议类型

协议定义

1.heartbeat,心跳协议

心跳用来维持长连接，当长时间没有心跳或者发送数据时，服务器将断开连接。 客户端目前设置参数： 心跳间隔：60秒 心跳超时次数：连续6次未收到

2.Request,客户端发送数据协议

字段	字节长度	类型	备注
encoding	2	String	前1位表示编码，默认1采用UTF-8。后一位为压缩方式，默认0，不压缩。
AccessToken	32	string	为用户登录后的token，验证token将通过此字段
systemCode	2	string	系统编号，ios用户端为11，android用户端为12
SourceID	8	string	渠道来源,标识推干广广市场
Userid	32	string	用户id
ClientID	20	string	客户端唯一一标识号;生生成规则待定
ClientVersion	4	Int	版本号。1.0.2对应值为1*10000*+*0*1000+2*1=10002
ServiceCode	6	string	服务/命令编号
sequence	4	int	数据序号

3.Response,服务器收到客户端发送的request后，返回的数据协议

字段	字节长度	类型	备注
encoding	2	string	前1位表示编码，默认1采用UTF-8。后一位为压缩方式，默认0，不压缩
ResponseCode	4	int	响应编号
ServiceCode	6	string	对应request的服务编码
accessToken	32	string	对应request的token

ClientID	20	string	对应request的clientID
sequence	4	int	对应request的sequence
isLast	1	boolean	是否最后一个消息，默认1，最后一个消息

4.Notify,服务器主动通知客户端消息协议

字段	字节长度	类型	备注
encoding	2	string	前1位表示编码，默认1采用UTF-8。后一位为压缩方式，默认0，不压缩
notifyCode	4	string	表示示notify类型,比比如订车车,车车库。。。
userId	32	string	用用户id
timestamp	8	long	消息推送时间戳
sequence	4	int	通知序号

5.ACK,客户端收到消息后回复协议

字段	字节长度	类型	备注
notifyCode	4	string	对应Notify的notifyCode
userId	32	string	用户id
sequence	4	int	对应Notify的sequence

6.Register,客户端注册通知消息协议

字段	字节长度	类型	备注
uid	32	string	用户id
sequence	4	int	注册序号

NotifyCode

NotifyCode	body	功能
1001	user.order.check结果	推送司机接单结果
2001	driver.order.getPush结果	有新的订单，推送订单给停车员
2002	车库推送消息内容	推送车库信息到停车员
2003	driver.order.getUserPos结果	推送用户位置到停车员

ServiceCode

ServiceCode	Action	功能
100101	user.order.check	检查订单状态
100001	user.park.request	下停车订单
100201	user.take.request	下取车订单
200001	driver.order.getPush	停车员获取订单
200002	driver.order.check	停车员检查订单状态
200003	driver.order.getUserPos	获取用户位置
200101	driver.account.updatePos	停车员上报位置

3.机制简介

1）用户处于登录态且有网络状态的情况下，向服务器注册长链接，同时维护心跳，心跳间隔10秒
2）每隔1小时检测一次心跳机制是否正常
3）收到notify类型消息，发送ack给服务器
4）解析服务器返回或推送数据，解析数据并回调上层注册的listener

4.Request task，tcp请求

1) request task 将请求构造成requestProtocol后发送，然后监听底层的回调，接收到回调后将response内容通过主线程回调
2) tcp请求传入请求类型和内容，回调可选
3) 超时和重试暂未实现

5.示例：

```
new RequestTask(context,
                ((CarKeyApplication) context.getApplicationContext()).getSocketService(),
                ServiceCode.USER_ORDER_CHECK,
                JsonUtil.toJson(checkOrder),
                new RequestTask.RequestTaskCallback() {
                    @Override
                    public void onSuccess(String response) {
                       ...
                    }

                    @Override
                    public void onFailed(int errorCode) {
                        ...
                    }
                }).run();
```