

水滴健康服务对外接口文档

版本	描述	时间	状态	作者
V1.0		2021-07-28	C	李志江
V1.1	回传身份证和手机号不需要脱敏直接加密	2021-08-02	M	李志江

C-创建；M-修改；D-删除；A-新增

1、概述

本文档用于电话医生的 api 接口对接。

2、数据传输

2.1、数据传输形式

数据采用 http post 传输，请求格式为 json，响应格式为 json。

2.2、数据加密

为了保证双方数据安全性，所有接口都需要按照签名规则进行加密，参考特殊说明。

2.3、幂等

- 如果订单已经存在，用相同单号重复提单。如果订单有效，返回该订单信息，不要返回失败；如果订单非有效，需要返回错误信息。
- 水滴业务侧也做了幂等，如果遇到回调超时，需要重试。

2.4、性能

平常情况下，接口应具备以下指标：

指标	描述
响应时间	99%在 200ms 以内
吞吐量	50qps

3、接口列表

3.1、下单接口

调用方：

水滴健康

接口说明：

用于接收下单数据

请求地址：

开发线地址：

请求 header

参数名称	参数类型	描述	必填	备注
uid	string	请求流水	是	请求流水，单次请求唯一，无业务含义，(作用类似 traceId)
bizType	string	业务类型	是	业务类型，见 bizType 取值
supplierNo	string	供应商编码	是	业务渠道(参与验证的合作方账号信息) 水滴提供

timestamp	long	请求时间戳	是	秒数
sign	string	签名	是	MD5(supplierNo+bizReqContent+bizType+timestamp+uid+key) 注意:bizReqContent为编码后的字符串, key 为验签密钥(水滴提供)

参数名称	参数类型	描述	必填	备注
bizOrderNo	string	订单号	是	健康业务订单号
userName	string	用户姓名 (通过 key 进行 AES 加密)	是	脱敏隐藏姓名的第二个字; 如果名字为三个字及以上, 则只保留首尾两个字, 隐藏中间部分
userPhone	string	手机号 (通过 key 进行 AES 加密)	是	传输脱敏电话客户手机号: 脱敏仅展示手机号后 4 位
effectTime	string	订单生效时间, 格式如: 2019-01-01 00:00:00	是	
invalidTime	string	订单截止日期, 格式如: 2019-12-31 23:59:59	是	

请求参数说明:

返回参数说明:

参数名称	参数类型	描述	必填	备注
code	Int	接口调用结果编码	是	0:成功 其他均为失败
msg	string	消息	是	
data	object		否	消息体
servNo	string	三方单号	否	成功时, 该值必填

请求参数示例:

```
{
  "bizOrderNo": "SDF000000007",
  "userName": "凯*尔",
  "phone": "*****0323",
  "effectTime": "2021-07-30 00:00:00",
  "invalidTime": "2022-07-30 00:00:00"
}
```

加密后参数参数:

```
{
  "bizResContent": "demoData"
}
```

返回响应结果参数 (加密的) :

```
{
  "bizResContent": "demoData"
}
```

返回响应结果解密后 (成功的) :

```
{
  "code": 0,
  "msg": "",
  "data": {
    "serviceOrderNo": "B1b320f78e65a47f4b93b5977d119cfb5"
  }
}
```

返回响应结果解密后 (失败的) :

```
{
  "msg": "该保单已经存在, 不能重复操作",
  "code": "-10000"
}
```

3.2、数据回传

调用方:

供给方

接口说明:

用于查询保单状态数据
请求地址:

开发线地址:
</sdb/vas/normalorder/v1/notify/{productNo}/callback>

productNo 由水滴健康提供

请求 header

参数名称	参数类型	描述	必填	备注
uid	string	请求流水	是	请求流水，单次请求唯一，无业务含义，(作用类似 traceId)
bizType	string	业务类型	是	业务类型，见 bizType 取值
supplierNo	string	供应商编码	是	业务渠道，水滴提供，(参与验证的合作方账号信息)
timestamp	long	请求时间戳	是	秒数
sign	string	签名	是	MD5(supplierNo+bizReqContent+bizType+timestamp+uid+key) 注意:bizReqContent为编码后的字符串，key 为验签密钥(水滴提供)

请求参数说明:

参数名称	参数类型	描述	必填	备注
bizOrderNo	string	订单号	是	健康业务订单号
servUseNo	String	三方流水号	是	三方流水号
callType	Integer	通话类型	是	0: 客户来电 1: 医生回电
userName	string	用户姓名 (通过 key 进行 AES 加密)	是	不需要脱敏

phone	string	手机号 (通过 key 进行 AES 加密)	是	不需要脱敏后
gender	Integer	性别	否	0:男 1:女
age	Integer	年龄	否	
startConsultingTime	string	咨询开始时间 格式: yyyy-MM-dd HH:mm:ss	是	
endConsultingTime	string	咨询结束时间 格式: yyyy-MM-dd HH:mm:ss	是	
talkTime	string	通话时长	是	单位: 秒
serviceStatus	Integer	服务状态	是	0:已完成服务 1:未服务 (服务开始, 客户挂断)
serviceResult	String	服务结果	否	处理类型
remainingTimes	Integer	剩余次数	是	
consultingDepartment	String	咨询科室	否	
customerAppeal	String	主诉	否	
medicalHistory	String	病史	否	
proposedDiagnosis	String	拟诊结果	否	诊断结果
handlingOpinions	String	处理意见	否	治疗建议
communicateContent	String	沟通内容	否	
fileUrls	List<String>	录音文件 url	否	(如果不支持 Sip 对接则为必传)

返回参数说明:

参数名称	参数类型	描述	必填	备注
code	Int	接口调用结果编码	是	0:成功 其他均为失败
msg	string	消息	是	
data	object		否	消息体

参数示例:

```
{
  "bizOrderNo": "demoData",
  "serviceBusinessNo": "张*忌",
  "userName": "demoData",
  "phone": "demoData",
  "gender": 1,
  "age": 1,
  "startConsultingTime": "2021-07-30 00:00:00",
  "endConsultingTime": "2021-07-30 00:06:00",
```

```
"talkTime":"demoData",
"serviceStatus":1,
"serviceResult":"demoData",
"remainingTimes":1,
"consultingDepartment":"demoData",
"customerAppeal":"demoData",
"medicalHistory":"demoData",
"proposedDiagnosis":"demoData",
"handlingOpinions":"demoData",
"communicateContent":"demoData",
"urls":[
  "https://img0.baidu.com/it/u=3498101689,3136759589&fm=253&fmt=auto&app=120&cf=JPEG?w=690&h=354"
]
```

加密后参数参数:

```
{
  "bizResContent":"demoData"
}
```

返回响应结果参数（加密的）：

```
{
  "bizResContent":"demoData"
}
```

返回响应结果解密后（成功的）：

```
{
  "msg":"成功",
  "code":"0"
}
```

返回响应结果解密后（失败的）：

```
{
  "msg":"失败",
  "code":"-10000"
}
```

4、特殊说明

4.1、bizType 取值

取值	描述	请求方
SERV_ORDER	服务出单	水滴
SERV_CANCEL	服务取消	水滴
SERV_USE	使用服务（三方回调）	三方
SERV_ORDER_QUERY	订单查询	水滴
SERV_BIND_USER	服务绑定	水滴

4.2、返回码

返回码	说明
0	成功
4000 ~ 4999 业务异常	
4001	请求频率限制，请 5s 后用相同参数重试
4004	订单号不存在
4015	签名 key 不存在
4016	加密错误
4017	解密错误
4023	签名错误

4024	bizType 参数错误
4025	参数错误
4026	此订单已取消
4027	此订单已过期
4028	此订单额度已用尽
4029	取消失败，此订单已取消过了
4030	客户身份证号有误
4031	三方下单失败
5000 ~ 5999 其他异常（如：系统内部异常），可进行重试【服务器错误类】	
5000	数据库错误
5999	系统错误

5、安全规范

5.1、加密解密

```
public static String aesEncrypt(String str, String key) {
    if (Strings.isNullOrEmpty(str) || Strings.isNullOrEmpty(key)) { return null; }
    try {
        Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");
        cipher.init(Cipher.ENCRYPT_MODE, new SecretKeySpec(key.getBytes("utf-8"),
"AES"));
        byte[] bytes = cipher.doFinal(str.getBytes("utf-8"));
        return Base64.encodeBase64String(bytes);
    } catch (Exception e) {
        throw new IllegalArgumentException("aes encrypt error", e);
    }
}
```

```
    }  
}  
  
public static String aesDecrypt(String str, String key) {  
    if (Strings.isNullOrEmpty(str) || Strings.isNullOrEmpty(key)) { return null; }  
    try {  
        Cipher cipher = Cipher.getInstance("AES/ECB/PKCS5Padding");  
        cipher.init(Cipher.DECRYPT_MODE, new SecretKeySpec(key.getBytes("utf-8"),  
"AES"));  
        byte[] bytes = Base64.decodeBase64(str);  
        bytes = cipher.doFinal(bytes);  
        return new String(bytes, "utf-8");  
    } catch (Exception e) {  
        throw new IllegalArgumentException("aes decrypt error", e);  
    }  
}
```

5.2、 验签

