**中山大学附属第一医院**

**短信服务采购需求**

1. **短信服务采购的必要性**

目前，中山大学附属第一医院短信发送服务即将到期，为保证医院业务不受影响，希望通过采购新一期短信发送服务，实现医院短信平台运营管理方式的整体升级优化，达到提高稳定性、高效性、安全性、便捷性，降低出错率、运营成本的目的，提高各科室企业、业务系统的整体工作效率。

1. **申请项目对医院短信沟通业务的支撑情况**

中山大学附属第一医院短信平台目前使用平台短信发送服务的医院各业务系统已经有85个，由于业务不断增多，业务短信发送需求日益增大，包含了发送量大的业务办理及通知等类型短信，预计2026-2028年每月短信发送量将持续增长。

本次项目申请采购短信套餐，并由提供套餐的通讯运营商建设2套短信发送服务平台（私有化部署平台+互联网SaaS平台），私有化部署平台需独立部署于医院的本地机房，保证数据存储本地化，提升数据安全性。

1. **申请采购短信发送项目预期的使用情况**

|  |  |
| --- | --- |
| **项目** | **数量** |
| 短信发送服务 | 9000万条 |
| 短信发送服务平台（私有化部署平台） | 1套 |
| 短信发送服务平台（互联网SaaS平台） | 1套 |

这批短信的使用周期预期2028年12月使用完毕。

1. **短信发送服务平台设计原则**

## 安全性和合规性

在建设短信系统时充分考虑采用成熟、可靠、保密、符合数据安全及合规管理要求的产品和技术措施，保证系统的安全。

## 成熟性和先进性

短信平台在系统建设中充分考虑成熟度和先进性，系统采用成熟的框架，能保证系统正常、稳定运行。同时，在系统设计时，考虑了模块化、跨平台应用等新技术，使系统具备国内同业领先的地位，保证投资的有效性和延续性。

## 可靠性和可用性

系统充分保证系统性能的可靠，保证其7×24小时连续、正常地工作。充分考虑系统故障时，数据的完整性和可靠性，平均故障修复时间小于1小时。系统运行开始，要有完备的数据备份与数据恢复策略，保证7\*24小时的工作状态。

## 接口程序可查性

由于企业业务系统发送短信经常涉及调用短信平台接口情况，某业务系统发生错误或接口程序错误均可能导致信息发送失败。故与业务系统接口程序方面应注意报错日志的管理，能随时区分出错误产生的位置，以供管理员查询，快速解决故障。

短信平台详细技术参数要求见附件1。

# 商务要求

本项目服务地点：中山大学附属第一医院本部。

付款方式：服务期为三年，款项分四期支付。短信发送服务每满9个月，依据甲方确认的短信发送数量及对应金额，进行一次合同款项的据实结算支付。甲方在收到乙方付款申请书以及按甲方确认金额开具的合法有效增值税普通发票后的30个自然日内，向乙方支付该次结算款项。

**附件1：短信平台技术参数要求**

# 功能要求：

## 短信平台操作功能要求

### 主页门户设计

可以按照需求定制化登录窗口，配置专属域名、企业名称和Logo，体现展现企业风貌。

### 统一管理

支持统一发送，统一各消息渠道的发送任务创建入口，消息渠道支持智能消息、短信、视频短信、空号过滤功能业务、短链功能业务、发送数据报表、5G消息、业务场景、微信公众号、APP推送。

（1）管理员可以通过用户管理，在帐户管理功能里新增子用户，包括对子帐号的权限设置。权限设置包括子帐号的导出下行记录、导出上行记录、业务启停、修改子号密码、搜索短信短信等功能。

业务审核功能：平台支持管理帐号对子帐号提交信息的审批权限。

短信发送：平台发送可以直接检测黑关键词功能，在发送界面可直接插入在平台生成的相关联短链接，并支持5G卡片短信发送。支持发送主题、发送内容、短信原文、高级设置，消息回落、发送对象、业务类型、签名、账号、发送时间创建发送任务，高级设置支持自定义短链类型，分为群发和个性化；

### 短信回复

针对操作员发送的短信信息，如有上行回复内容，可通过系统上行记录查看了解相应手机号的回复内容。包括但不限于：接收时间、通道号码、手机号码、上行回复的内容等。后续可与企业业务系统对接，实现上行回复内容的同步展现。

### 统计报表

支持统计报表功能：

短信帐号报表统计：短信账号使用的统计日报表跟月报表，可以【导出】数据；支持导出日报表或月报表形式，包括支持导出：提交量、提交率、通道成功、通道失败、通道未知、上行数、回执率、成功率、上行率、运营商等数据。

### 系统管理

操作员可通过用户管理操作对自己的账号信息进行修改，和部门人员查看、修改密码、联系人、联系号码等操作。

支持角色管理：

（1）创建操作员的不同类型角色以及为不同类型角色分配不同的权限；

（2）在创建操作员时可按照分配对应的角色赋予对应的权限；

（3）每个角色对应角色管理模块分配的权限；

支持高级安全设置：

（1）企业对系统使用安全的管控，保证企业操作员安全登陆系统，系统可以根据客户侧实际使用情况，将操作员与IP地址进行绑定或启用动态口令，可以限制其操作员只能在指定IP地址的电脑上进行操作；

（2）启用动态口令，操作员输入登录账户和密码成功后，主动获取手机短信验证码，输入对应的验证码后，才可以登陆到系统首页正常使用平台功能，增强型登陆安全设置支持一种或多种并用的安全登录模式；

（3）支持平台登录地址IP白名单，加白登录。

## 短信平台业务系统接口功能要求

符合并实现《短信平台标准接口文档》（附件）的技术要求。

附件：

**短信平台标准接口文档**

1. 前言

本协议基于HTTP服务(支持HTTPS)，使用POST请求方式，请求和应答均为JSON格式数据。

字段命名方式：驼峰法。

统一请求和响应编码：UTF-8

统一请求Header内容：Content-Type: application/json

请使用接口网关地址替换文档中的服务器地址：http(s)://{address:port}/sms

sign参数计算规则：多个指定参数值组合成字符串后计算MD5 32位小写结果

要求：MD5(userName + timestamp + MD5(password))

假设：userName(帐号名)=test

password(帐号密码)=123

timestamp=1596254400000

计算：MD5(password)=202cb962ac59075b964b07152d234b70

组合字符串：test1596254400000202cb962ac59075b964b07152d234b70

sign结果：MD5(组合字符串)=e315cf297826abdeb2092cc57f29f0bf

1. **短信批量发送接口**
   1. 调用地址

HPPT地址：http://{address:port}/sms**/api/sendMessageMass**

HPPTS地址：https://{address:port}/sms**/api/sendMessageMass**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| content | String | 是 | 短信内容 |
| phoneList | [Array] | 是 | 发送手机号码，JSON数组格式。  最大数量不得超过10000个号码，系统将自动去除重复号码。 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |
| sendTime | String | 否 | 短信定时发送时间，格式：yyyy-MM-dd HH:mm:ss。  定时时间限制15天以内。 |
| extcode | String | 否 | 可选，附带通道扩展码 |
| callData | String | 否 | 用户回传数据，最大长度64。  用户若传递此参数将在回执推送时回传给用户。 |

* 1. **响应结果**

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| msgId | Long | 当code=0时，系统返回唯一消息Id |
| smsCount | Integer | 当code=0时，系统返回消耗计费总数 |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/sendMessageMass

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"content": "【签名】您的验证码是123456",

"phoneList": ["13500000001", "13500000002", "13500000003"],

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf"

}

响应结果：

{

"code": 0,

"message": "处理成功",

"msgId": 123456,

"smsCount": 3

}

1. **短信一对一发送接口**
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/sendMessageOne**

HTTPS地址：https://{address:port}/sms**/api/sendMessageOne**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| messageList | [Array] | 是 | 数组形式，包含多个JSON对象，对象参数见下表。  每个JSON对象包含短信内容和号码数据，最大1000个号码。 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |
| sendTime | String | 否 | 短信定时发送时间，格式：yyyy-MM-dd HH:mm:ss。  定时时间限制15天以内。 |

messageList由多个JSON对象构成的JSON数组，具体参数列表：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| phone | String | 是 | 发送手机号码 |
| content | String | 是 | 短信内容 |
| extcode | String | 否 | 可选，附带通道扩展码 |
| callData | String | 否 | 用户回传数据，最大长度64。  用户若传递此参数将在回执推送时回传给用户。 |

* 1. **响应结果**

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| data | [Array] | 当code=0时，系统返回处理结果的数组对象集合，对象参数见下表。 |
| smsCount | Integer | 当code=0时，系统返回消耗此次请求的计费总数 |

data由多个JSON对象构成的JSON数组，具体参数列表：

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| phone | String | 发送手机号码 |
| msgId | Long | 当code=0时，系统返回唯一消息Id |
| smsCount | Integer | 当code=0时，系统返回此号码的计费数 |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/sendMessageOne

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"messageList": [

{

"phone": "13500000001",

"content" : "【签名】尊敬的张先生，本次共消费211.45元"

},

{

"phone": "13500000002",

"content" : "【签名】尊敬的林女士，本次共消费78.00元"

}

],

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf"

}

响应结果：

{

"code": 0,

"message": "处理成功",

"smsCount": 2,

"data": [

{

"code": 0,

"message": "处理成功",

"msgId": 11600001,

"phone": "13500000001",

"smsCount": 1

},

{

"code": 0,

"message": "处理成功",

"msgId": 11600002,

"phone": "13500000002",

"smsCount": 1

}

]

}

1. 回执状态推送接口
   1. 调用地址

地址：客户需向我司提交接收回执状态地址，由平台主动推送回执状态数据

推送请求方法：POST

* 1. 推送请求包头定义

Content-Type: application/json;charset=utf-8

* 1. 请求参数

推送数据为JSON数组形式，每次推送不大于2000条。推送字段如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| msgId | Long | 是 | 消息id，对应发送成功时系统响应的msgId |
| phone | String | 是 | 手机号码 |
| status | String | 是 | 回执状态，DELIVRD成功，其他失败 |
| receiveTime | String | 是 | 回执时间，格式：yyyy-MM-dd HH:mm:ss |
| smsCount | Integer | 是 | 此发送号码的计费条数 |
| callData | String | 否 | 用户回传数据，如果提交时有传递此参数将原样推送带回 |
| diffStatus | [Array] | 否 | 当长短信拆分发送后回执状态码不一致时，会将多个片段状态码传递此参数。字符串数组格式，例如：['DELIVRD', 'MK:0001'] |

* 1. **响应结果**

正常响应HTTP状态码200即可。非200状态码将转换为客户获取形式

* 1. **推送请求示例**

[

{

"msgId": 11600001,

"phone": "13500000001",

"receiveTime": "2020-06-09 11:10:32",

"status": "DELIVRD",

"smsCount": 1

},

{

"msgId": 11600002,

"phone": "13500000002",

"receiveTime": "2020-06-09 11:10:32",

"status": "FAILURE",

"smsCount": 1

}

]

1. 上行回复推送接口
   1. 调用地址

地址：客户需向我司提交接收上行回复地址，由平台主动推送上行回复数据

推送请求方法：POST

* 1. 推送请求包头定义

Content-Type: application/json;charset=utf-8

* 1. 请求参数

推送数据为JSON数组形式，每次推送不大于2000条。推送字段如下：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| content | String | 是 | 上行回复内容 |
| phone | String | 是 | 手机号码 |
| receiveTime | String | 是 | 回执时间，格式：yyyy-MM-dd HH:mm:ss |
| destId | String | 否 | 通道端口号 |
| msgId | Long | 否 | 短信发送提交时响应的消息id |
| callData | String | 否 | 用户回传数据，如果提交时有传递此参数将原样推送带回 |

* 1. **响应结果**

正常响应HTTP状态码200即可。非200状态码将转换为客户获取形式

* 1. **推送请求示例**

[

{

"content": "好的, 已收到",

"destId": "106203069598",

"phone": "13500000001",

"receiveTime": "2020-06-09 11:10:32"

},

{

"content": "OK",

"phone": "13500000002",

"receiveTime": "2020-06-09 11:10:32"

}

]

1. **回执状态获取接口**
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/getReport**

HTTPS地址：https://{address:port}/sms**/api/getReport**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

此接口每次请求间隔时间不得小于30秒，如果获取条数为limit(默认2000条)表示还有回执未获取，可立即再次请求获取回执。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |
| limit | Integer | 否 | 最大获取数，默认2000，可选范围10~10000 |

* 1. **响应结果**

响应为JSON形式，每次获取不大于limit(默认2000条)，已获取数据不会被再次获取到。

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| data | [Array] | 获取的回执列表。JSON数组形式，具体字段如下 |

data包含推送字段如下（与[4.3](#_回执状态推送接口)推送参数一致）

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| msgId | Long | 是 | 消息id，对应发送成功时系统响应的msgId |
| phone | String | 是 | 手机号码 |
| status | String | 是 | 回执状态，DELIVRD成功，其他失败 |
| receiveTime | String | 是 | 回执时间，格式：yyyy-MM-dd HH:mm:ss |
| smsCount | Integer | 是 | 此发送号码的计费条数 |
| callData | String | 否 | 用户回传数据，如果提交时有传递此参数将原样推送带回 |
| diffStatus | [Array] | 否 | 当长短信拆分发送后回执状态码不一致时，会将多个片段状态码传递此参数。字符串数组格式，例如：['DELIVRD', 'MK:0001'] |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/getReport

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf"

}

响应结果：

{

"code": 0,

"message": "处理成功",

"data": [

{

"msgId": 11600001,

"phone": "13500000001",

"receiveTime": "2020-06-09 11:10:32",

"status": "DELIVRD",

"smsCount": 1

},

{

"msgId": 11600002,

"phone": "13500000002",

"receiveTime": "2020-06-09 11:10:32",

"status": "FAILURE",

"smsCount": 1

}

]

}

1. **上行回复获取接口**
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/getUpstream**

HTTPS地址：https://{address:port}/sms**/api/getUpstream**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

此接口每次请求间隔时间不得小于30秒，如果获取条数为limit(默认2000条)表示还有上行未获取，可立即再次请求获取上行数据。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |
| limit | Integer | 否 | 最大获取数，默认2000，可选范围10~10000 |

* 1. **响应结果**

响应为JSON形式，每次获取不大于limit(默认2000条)，已获取数据不会被再次获取到。

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| data | [Array] | 获取的上行列表。JSON数组形式，具体字段如下 |

data包含推送字段如下（与5.4推送参数一致）

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| content | String | 是 | 上行回复内容 |
| phone | String | 是 | 手机号码 |
| receiveTime | String | 是 | 回执时间，格式：yyyy-MM-dd HH:mm:ss |
| destId | String | 否 | 通道端口号 |
| msgId | Long | 否 | 短信发送提交时响应的消息id |
| callData | String | 否 | 用户回传数据，如果提交时有传递此参数将原样推送带回 |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/getUpstream

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf"

}

响应结果：

{

"code": 0,

"message": "处理成功",

"data": [

{

"content": "好的, 已收到",

"destId": "106203069598",

"phone": "13500000001",

"receiveTime": "2020-06-09 11:10:32"

},

{

"content": "OK",

"phone": "13500000002",

"receiveTime": "2020-06-09 11:10:32"

}

]

}

1. 查询余额接口
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/getBalance**

HTTPS地址：https://{address:port}/sms**/api/getBalance**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |

* 1. **响应结果**

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| balance | Long | 当code=0时，系统返回帐号短信余额 |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/getBalance

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf"

}

响应结果：

{

"code": 0,

"message": "处理成功",

"balance": 967793

}

1. 提交短信模板接口
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/createTemplate**

HTTPS地址：https://{address:port}/sms**/api/createTemplate**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |
| content | String | 是 | 模板内容，内容文本中变量符：{%变量%} |
| type | Integer | 否 | 模板类型，1 - 精准目标，2 - 模糊模板。默认为精确模板 |
| matchPercent | Integer | 否 | 模糊匹配百分比，当type=2时必填，有效范围：60-100 |
| expireDate | String | 否 | 失效日期，格式：yyyy-MM-dd。留空为永久有效 |

* 1. **响应结果**

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| templateId | Integer | 短信模板Id |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/createTemplate

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf",

"content": "【签名】您的验证码是{%变量%} "

}

响应结果：

{

"code": 0,

"message": "处理成功",

"templateId": 336

}

1. 查询短信模板接口
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/queryTemplates**

HTTPS地址：https://{address:port}/sms**/api/queryTemplates**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

此接口每次请求间隔时间不得小于60秒。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |
| templateId | Integer | 否 | 短信模板Id |

* 1. **响应结果**

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| data | [Array] | JSON对象数组, 返回已生效的短信模板列表。如果指定查询模板Id，只返回此模板已生效内容，不生效则返回空数组。  templateId: 模板Id  content: 模板内容  type: 模板类型，1 - 精准目标，2 - 模糊模板  matchPercent: 模板匹配度，type=2时必填 |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/queryTemplates

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf"

}

响应结果：

{

"code": 0,

"message": "处理成功",

"data": [

{ "templateId": 1, "content": "【签名】您的验证码是{%变量%}", "type": 1 },

{ "templateId": 2, "content": "【签名】亲爱的顾客, 您本次共消费12元, 感谢光临", "type": 1, "matchPercent": 80 }

]

}

1. **报备签名接口**
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/addSignature**

HTTPS地址：https://{address:port}/sms**/api/addSignature**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

此接口用于提交报备短信签名，提交后的签名需经过审核方可生效。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |
| signatureList | [Array] | 是 | 报备签名列表，JSON数组格式。数组内填写报备签名，包含完整“【】”符号，例如：["【签名1】", "【签名2】"] |

* 1. **响应结果**

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/addSignature

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf",

"signatureList": ["【签名1】", "【签名2】"]

}

响应结果：

{

"code": 0,

"message": "处理成功"

}

1. **查询签名接口**
   1. 调用地址

HTTP地址：http://{address:port}/sms**/api/querySignature**

HTTPS地址：https://{address:port}/sms**/api/querySignature**

请求方法：POST

* 1. 请求包头定义

Accept: application/json

Content-Type: application/json;charset=utf-8

* 1. 请求参数

此接口每次请求间隔时间不得小于30秒，可查询帐号可用的所有短信签名。

|  |  |  |  |
| --- | --- | --- | --- |
| 参数名 | 类型 | 必填 | 说明 |
| userName | String | 是 | 帐号用户名 |
| timestamp | Long | 是 | 当前时间戳，精确到毫秒。  例如2020年8月1日12:00:00 时间戳为：1596254400000 |
| sign | String | 是 | 由以下参数值组合成字符串并计算MD5值，参考[详细规则](#_前言)  计算：MD5(userName + timestamp + MD5(password)) |

* 1. **响应结果**

|  |  |  |
| --- | --- | --- |
| 参数名 | 类型 | 说明 |
| code | Integer | 处理结果，0为成功，其他失败，详细参考[响应状态码](#_响应状态码列表) |
| message | String | 处理结果描述 |
| data | [Array] | 可用签名列表，JSON数组格式。  例如：["【签名1】", "【签名2】"] |

* 1. **请求示例**

发送请求：

POST http://{address:port}/sms/api/querySignature

Accept: application/json

Content-Type: application/json;charset=utf-8

{

"userName": "test",

"timestamp": 1596254400000,

"sign": "e315cf297826abdeb2092cc57f29f0bf"

}

响应结果：

{

"code": 0,

"message": "处理成功"

"data": ["【签名1】", "【签名2】"]

}

1. **响应状态码列表**

|  |  |
| --- | --- |
| 状态码 | 说明 |
| 0 | 处理成功 |
| 1 | 帐号名为空 |
| 2 | 帐号名或密码鉴权错误 |
| 3 | 帐号已被锁定 |
| 4 | 此帐号业务未开通 |
| 5 | 帐号余额不足 |
| 6 | 缺少发送号码 |
| 7 | 超过最大发送号码数 |
| 8 | 发送消息内容为空 |
| 9 | 无效的RCS模板ID |
| 10 | 非法的IP地址，提交来源IP地址与帐号绑定IP不一致 |
| 11 | 24小时发送时间段限制 |
| 12 | 定时发送时间错误或超过15天 |
| 13 | 请求过于频繁，每次获取数据最小间隔为30秒 |
| 14 | 错误的用户扩展码 |
| 16 | 时间戳差异过大，与系统时间误差不得超过5分钟 |
| 18 | 帐号未进行实名认证 |
| 19 | 帐号未开放回执状态 |
| 22 | 缺少必填参数 |
| 23 | 用户帐号名重复 |
| 24 | 用户无签名限制 |
| 25 | 签名需要包含【】符 |
| 50 | 缺少模板标题 |
| 51 | 缺少模板内容 |
| 52 | 模板内容不全 |
| 53 | 不支持的模板帧类型 |
| 54 | 不支持的文件类型 |
| 97 | 此链接不支持GET请求 |
| 98 | HTTP Content-Type错误, 请设置Content-Type: application/json |
| 99 | 错误的请求JSON字符串 |
| 500 | 系统异常 |