

快信通 3 短信平台接口规范

版本 1.2

上海摩道网络科技有限公司

目录

1. 接口地址.....	3
1.1. 接口地址.....	3
1.2. 请求方式.....	3
1.3. 编码.....	3
1.4. 请求参数.....	3
1.5. 关于 JSON 格式.....	4
1.6. 密码安全.....	4
2. 身份认证.....	4
2.1. 请求参数.....	4
2.2. 请求样例.....	5
2.3. 响应消息.....	5
2.4. 响应样例.....	5
3. 发送短信.....	5
3.1. 请求参数.....	5
3.2. 请求样例.....	6
3.3. 响应消息.....	6
3.4. 响应样例.....	7
4. 状态报告.....	7
4.1. 获得方式.....	7
4.2. 推送地址.....	7
4.3. 推送消息的编码.....	7
4.4. 推送消息的结构.....	8
4.5. 状态报告消息的结构.....	8
4.6. 推送消息样例.....	8
4.7. 客户响应.....	9
5. 上行短信.....	9
5.1. 获得方式.....	9
5.2. 推送地址.....	9
5.3. 推送消息的编码.....	9
5.4. 推送消息的结构.....	9
5.5. 上行短信的结构.....	10
5.6. 推送消息样例.....	10
5.7. 客户响应.....	10
6. 附录.....	11
6.1. 加密规则.....	11
6.2. 状态码表.....	11

1. 接口地址

1.1. 接口地址

快信通 3 短信平台使用一个统一的地址，接收不同的业务请求。

```
http://121.40.152.72:8180/webservice/http/service
```

1.2. 请求方式

请用 POST 方式提交请求。

```
POST
```

1.3. 编码

请求和响应都使用相同的编码。

```
UTF-8
```

1.4. 请求参数

所有请求都是由两个参数构成：op 和 message。

参数	说明
op	命令字；具体值由业务请求决定。
message	参数体；具体值由业务请求决定；是 JSON 格式的字符串。

第一个参数 op 是命令字，具体值由业务决定（例如，在身份认证请求中 op 值为 login，而在发送短信请求中 op 值为 sendSm）。

第二个参数 message 是参数体，是一个数据结构（对象）经过串行化后的 JSON 格式字符串。

1.5. 关于 JSON 格式

例如，下面这个数据结构（对象）是这样的：

```
User {  
    String name = "Jim";  
    int age = 18;  
}
```

这个数据结构（对象）经过串行化后的 JSON 字符串就是：

```
{  
  "name": "Jim",  
  "age": 18  
}
```

目前有很多的开源工具可以将数据结构（对象）转换成 JSON 字符串。

1.6. 密码安全

每个请求都含有身份认证信息，为了保障客户的信息安全，防止原始密码泄露，客户端需要将密码加密后再传输；同时，为了防止请求消息被截获，避免被非法地重复利用，客户端需要在加密的时候使用不重复的时间戳。

2. 身份认证

2.1. 请求参数

参数	说明
cid	机构编号
uid	用户登录名
pwd	密码加密后的密文（加密办法请参见附录）
timestamp	时间戳； 18 位字符串；格式是 yyyyMMddHHmmssNNNN；前面是 14 位的时间（精确到秒），再追加 4 位随机数字，总共 18 位； 请注意，平台要求每次请求的时间戳都是不重复的。

2.2. 请求样例

参数	值
op	login
message	{ "cid": "12345678", "uid": "test", "pwd": "0723d239c0c8007f09aaaeabb2807e977", "timestamp": "201501242011311000" }

2.3. 响应消息

参数	说明
stateCode	状态码；0 表示操作成功，其他均为错误或异常；具体意义请参见附录状态码表。
message	一串文本；有助于理解操作的结果。
smBalance	短信余额
mmBalance	彩信余额

2.4. 响应样例

{ "stateCode": 0, "message": "操作成功", "smBalance": 0, "mmBalance": 0 }
--

3. 发送短信

3.1. 请求参数

参数	说明
cid	机构编号
uid	用户登录名
pwd	密码（加密后的密文；加密办法请参见附录）

timestamp	时间戳： 18 位字符串，格式是 yyyyMMddHHmmssNNNN；前面是 14 位的时间（精确到秒），再追加 4 位随机数字，总共 18 位； 请注意，平台要求每次请求的时间戳都是不重复的。
mobiles	目标号码；数组格式；平台支持每次最多 10000 个号码。
content	短信内容；平台支持最多 1000 字。
atTime	定时发送时间； 如果是立即发送，则可以不填； 如果要填，其格式是 yyyy-MM-dd HH:mm:ss（19 位长度）； 例如 2015-04-01 12:00:00
dailyBeginTime	每日限发时段的起始时间； 可选项，可以不填； 如果填写，其格式是 HH:mm:ss(8 位长度)； 例如，如果希望早上九点半开始发，则该参数可填写为 09:30:00； 该参数最小值是 00:00:00。
dailyEndTime	每日限发时段的截止时间； 可选项，可以不填； 如果填写，其格式是 HH:mm:ss(8 位长度)；例如，如果希望下午六点后不要发，则该参数可填写为 18:00:00； 该参数最大值是 23:59:59。

3.2. 请求样例

参数	值
op	sendSm
message	{ "cid": "12345678", "uid": "test", "pwd": "0723d239c0c8007f09aaebbb2807e977", "timestamp": "201501242011311000", "mobiles": ["13708600860", "18930612517"], "content": "周末参加不了培训了", "atTime": "", "dailyBeginTime": "08:00:00", "dailyEndTime": "18:00:00" }

3.3. 响应消息

参数	值
stateCode	状态码；0 表示操作成功，其他均为错误或异常；具体意义请参见附录状态码表。

message	一串文本；有助于理解操作的结果。
token	发送标识；请保存这个数据，该数据用于匹配状态报告；每一次发送请求，无论含有多少个号码，只会返回一个 token；每次请求都会获得新的 token。
fee	本次发送产生的费用；费用与号码数、内容长度有关系。

3.4. 响应样例

```
{
  "stateCode":0,
  "message":"操作成功",
  "token":376,
  "fee":1
}
```

4. 状态报告

4.1. 获得方式

快信通 3 平台采用推送（PUSH）机制，将状态报告通过 HTTP 消息推送至客户处。客户需要提供一个 HTTP 服务，能够接收并处理消息。

4.2. 推送地址

客户需要提交一个 URL 地址给快信通 3 平台的维护人员。例如下列地址

```
http://xxx.xxx.xxx:8080/receive
http://xxx.xxx.xxx:8080/receive.php
http://xxx.xxx.xxx:8080/receive.jsp
http://xxx.xxx.xxx:8080/receive.asp
```

4.3. 推送消息的编码

UTF-8

4.4. 推送消息的结构

参数	说明
uid	推送账号；由客户提交、设置。
pwd	推送密码（明文）；由客户提交、设置。
type	推送数据的类型；推送状态报告时，固定为 report。
jsonData	推送数据；将多个状态报告以 JSON 格式封装；具体请参见下一节的详细说明。

4.5. 状态报告消息的结构

参数	说明
token	发送标记；该数据与发送短信获得的 token 值有对应关系。
mobile	一个号码
resultCode	状态码；具体值由网关决定，常见的有 0、DELIVER 等。
success	状态码是否是成功

4.6. 推送消息样例

下面这个样例消息是快信通 3 平台向客户推送了两个状态报告：

```
{
  uid="abc",
  pwd="123",
  type="report",
  jsonData=[
    {
      "token":7331,
      "mobile":"13500000000",
      "resultCode":"0",
      "success":true
    },
    {
      "token":8469,
      "mobile":"13700000000",
      "resultCode":"2:46",
      "success":false
    }
  ]
}
```



```
}
```

4.7. 客户响应

如果客户正确接收并处理了消息，要返回字符串 0。

```
0
```

5. 上行短信

5.1. 获得方式

快信通 3 平台采用推送（PUSH）机制，将上行短信通过 HTTP 消息推送至客户处。客户需要提供一个 HTTP 服务，能够接收并处理消息。

5.2. 推送地址

客户需要提交一个 URL 地址给快信通 3 平台的维护人员。例如下列地址

```
http://xxx.xxx.xxx:8080/receive
http://xxx.xxx.xxx:8080/receive.php
http://xxx.xxx.xxx:8080/receive.jsp
http://xxx.xxx.xxx:8080/receive.asp
```

5.3. 推送消息的编码

```
UTF-8
```

5.4. 推送消息的结构

参数	说明
uid	推送账号；由客户提交、设置。

pwd	推送密码（明文）；由客户提交、设置。
type	推送数据的类型；推送上行短信时，固定为 mo。
jsonData	推送数据；将多个状态报告以 JSON 格式封装；具体请参见下一节的详细说明。

5.5. 上行短信的结构

参数	说明
mobile	手机号码（上行短信的来源手机号码）
content	短信内容
smgRecvDate	网关收到该上行短信的时间；格式 yyyy-MM-dd HH:mm:ss

5.6. 推送消息样例

下面这个样例消息是快信通 3 平台向客户推送了两个上行短信：

```
{
  uid="abc",
  pwd="123",
  type="mo",
  jsonData=[
    {
      "mobile":"13500000000",
      "content":"你好吗",
      "smgRecvDate":"2015-01-01 12:00:00"
    },
    {
      "mobile":"13700000000",
      "content":"我不好",
      "smgRecvDate":"2015-01-01 12:00:00"
    }
  ]
}
```

5.7. 客户响应

如果客户正确接收并处理了消息，要返回字符串 0。

```
0
```

6. 附录

6.1. 加密规则

密码密文字符串 = MD5（原始密码明文字符串 + 时间戳字符串）

【注】 计算出的 MD5 值是 32 位小写字母串。

6.2. 状态码表

状态码	意义
0	操作成功
1	系统忙（未知错误）；通常是平台维护，或者发生内部错误。
100	不支持的操作；提交了错误的 op 值。
110	提交了无效的 JSON 数据
120	缺少参数
130	不支持 GET 请求
200	认证错
210	请求频率过快
220	无效来源（请求方 IP 被平台拒绝）
230	无效的时间戳；格式不对，或者提交了重复的时间戳。
240	认证失败
250	账户已停用
260	机构已停用
300	号码数不正确；可能数量为 0，或者超过了平台限制。
310	包含无效号码
320	内容长度不正确；可能长度为 0，或超过了平台限制。
330	定时时间不正确
340	每日限发时段不正确
350	提交的 token 数量不正确；可能超过了平台限制。
360	提交的 token 值不正确
400	余额不足