**消息中心系统**

**接口文档**

证大爱特金融信息服务有限公司

2016年08月

版本历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 部门 | 日期 | 备注 |
| V1.0 | 高久成 | IT技术部 | 2016年08月 | 版本定义:  新增：推送接口  1收集设备  2消息推送 |
| V1.0 | 赖武法 | IT技术部 | 2016年08月 | 新增：消息中心接口  1.发送消息  2查询未读消息个数  3.读取消息  4获取消息内容  5获取分页消息内容 |
| V1.1 | 高久成 | IT技术部 | 2016年11月 | 新增：短信相关接口  1发送验证码短信  2发送通知短信  3发送营销短信  4验证码验证 |
| V1.2 | 高久成 | IT技术部 | 2017年02月 | 新增：访问日志接口  1.记录访问日志 |

目录

[1. 概述 5](#_Toc474413649)

[2. 系统架构 5](#_Toc474413650)

[3. 接口概述 5](#_Toc474413651)

[3.1 通信方式 5](#_Toc474413652)

[3.2 基本数据类型 5](#_Toc474413653)

[3.3 结构类型 6](#_Toc474413654)

[3.4 符号定义 6](#_Toc474413655)

[4. 推送接口 6](#_Toc474413656)

[4.1 收集设备接口 6](#_Toc474413657)

[4.1.1 接口描述 6](#_Toc474413658)

[4.1.2 请求参数 6](#_Toc474413659)

[4.1.3 请求结果 7](#_Toc474413660)

[4.2 消息推送接口 7](#_Toc474413661)

[4.2.1 接口描述 7](#_Toc474413662)

[4.2.2 请求参数 8](#_Toc474413663)

[4.2.3 请求结果 8](#_Toc474413664)

[5. 消息中心接口 9](#_Toc474413665)

[5.1 发送消息 9](#_Toc474413666)

[5.1.1 接口描述 9](#_Toc474413667)

[5.1.2 请求参数 9](#_Toc474413668)

[5.1.3 请求结果 10](#_Toc474413669)

[5.2 未读消息个数 11](#_Toc474413670)

[5.2.1 接口描述 11](#_Toc474413671)

[5.2.2 请求参数 11](#_Toc474413672)

[5.2.3 请求结果 11](#_Toc474413673)

[5.3 读取消息 12](#_Toc474413674)

[5.3.1 接口描述 12](#_Toc474413675)

[5.3.2 请求参数 12](#_Toc474413676)

[5.3.3 请求结果 13](#_Toc474413677)

[5.4 获取消息详情内容 13](#_Toc474413678)

[5.4.1 接口描述 13](#_Toc474413679)

[5.4.2 请求参数 13](#_Toc474413680)

[5.4.3 请求结果 14](#_Toc474413681)

[5.5 获取消息列表-分页 15](#_Toc474413682)

[5.5.1 接口描述 15](#_Toc474413683)

[5.5.2 请求参数 15](#_Toc474413684)

[5.5.3 请求结果 16](#_Toc474413685)

[6. 短信接口 17](#_Toc474413686)

[6.1 发送验证码短信 17](#_Toc474413687)

[6.1.1 接口描述 17](#_Toc474413688)

[6.1.2 请求参数 17](#_Toc474413689)

[6.1.3 请求结果 18](#_Toc474413690)

[6.2 发送通知短信 18](#_Toc474413691)

[6.2.1 接口描述 18](#_Toc474413692)

[6.2.2 请求参数 18](#_Toc474413693)

[6.2.3 请求结果 19](#_Toc474413694)

[6.3 发送营销短信 19](#_Toc474413695)

[6.3.1 接口描述 19](#_Toc474413696)

[6.3.2 请求参数 19](#_Toc474413697)

[6.3.3 请求结果 20](#_Toc474413698)

[6.4 验证码验证 21](#_Toc474413699)

[6.4.1 接口描述 21](#_Toc474413700)

[6.4.2 请求参数 21](#_Toc474413701)

[6.4.3 请求结果 21](#_Toc474413702)

[7. 访问日志接口 22](#_Toc474413703)

[7.1.1 记录访问日志 22](#_Toc474413704)

[7.1.2 接口描述 22](#_Toc474413705)

[7.1.3 请求参数 22](#_Toc474413706)

[7.1.4 请求结果 23](#_Toc474413707)

# 概述

证大财富内部核心系统，统一提供相关请求功能。

# 系统架构

# 接口概述

定义

核心接入规范明确各接入系统功能调用必须遵循的请求和响应参数。

## 通信方式

基于hessian

## 基本数据类型

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **类型缩写** | **类型描述** | **参考** |
| **1** | a | 字母a－z | A、a、ab |
| **2** | n | 数字0－9 | 1、12、-1 |
| **3** | s | 特殊字符 | @、# |
| **4** | an | 字母和数字字符 | Word2 |
| **5** | ans | 字母、数字和特殊字符 | Hell\_wold1 |
| **6** | yyyy | 4位数年份 | 2010（2010年） |
| **7** | MM | 年份中的月份 | 03（3月份） |
| **8** | DD | 月份中的日期 | 25（25号） |
| **9** | hh | 小时（24格式） | 15（15点） |
| **10** | mm | 分钟 | 30（30分） |
| **11** | ss | 秒 | 59（59秒） |
| **12** | datetime | 日期时间，格式为 yyyy-MM-DD hh:mm:ss | 2010-03-25 15:30:59 |
| **13** | time | 时间，new Date().getTime()的值 | 1412751060848 |

## 结构类型

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **类型缩写** | **类型描述** | **参考** |
| **1** | Array | 数组 |  |
| **2** |  |  |  |

## 符号定义

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **符号缩写** | **符号性质** | **符号说明** |
| **1** | M | 强制域(Mandatory) | 此域必选，否则将被认为格式出错 |
| **2** | C | 条件域(Conditional) | 此域在一定条件下必选 |
| **3** | O | 选用域(Optional) | 此域由发送方自选 |

# 推送接口

## 收集设备接口

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 收集设备接口 |
| 接口描述 | 收集设备tokenId |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| cellphoneNum | 手机号 | String | 20 | M |  |
| deviceId | 设备号 | String | 100 | M |  |
| deviceType | 设备类型 | String | 1 | M | 规则：  1:Andriod 2:IOS |

例子：

MsgDeviceCollectDto collectDto = **new** MsgDeviceCollectDto();  
collectDto.setDeviceType(**"2"**);  
collectDto.setCellphoneNum(**"13586623006"**);  
collectDto.setDeviceId(**"6951c73487fyre8f44273c"**);  
MessageResultDto resultDto = **msgPushFacadeService**.collect(collectDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDtoresult = msgPushFacadeService.collect(collectDto);  
**if** (result.isSuccess()) {  
 System.***out***.println(**"code="**+ result.getCode();

System.***out***.println(**"msg="**+ result.getMsg();  
} **else** {  
 System.***out***.println(**"错误消息="**+ result.getMsg());  
}

## 消息推送接口

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 消息推送接口 |
| 接口描述 | 推送消息到用户手机 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| serialNo | 请求流水号 | String | 50 | M | 商户请求流水号，不能重复 |
| phones | 手机号列表 | List<String> |  | C | 非全员推送必须 |
| pushTime | 推送时间 | Date |  | C | 推送时间 |
| isInstant | 是否立即推送 | boolean |  | M | true:是 false:否 |
| isAllpush | 是否全员推送 | boolean |  | M | true:是 false:否 |
| title | 通知标题 | String | 100 | M |  |
| content | 通知内容 | String | 500 | M |  |
| data | 消息内容 | String | 500 | O |  |

例子：

MsgPushDto pushDto = **new** MsgPushDto();

pushDto.setSerialNo(**"2016081107264875500900"**);  
pushDto.setPushTime(**new** Date());  
pushDto.setTitle(**"测试通知标题"**);  
pushDto.setContent(**"测试通知内容"**);  
pushDto.setIsInstant(**false**);  
pushDto.setIsAllpush(**false**);  
pushDto.setData(**"透传内容"**);  
List<String> phoneList = **new** ArrayList<>();  
phoneList.add(**"18650309360"**);  
phoneList.add(**"18650309362"**);  
pushDto.setPhones(phoneList);

MessageResultDto resultDto = **msgPushFacadeService**.push(pushDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDtoresult = msgPushFacadeService.collect(pushDto);  
**if** (result.isSuccess()) {  
 System.***out***.println(**"code="**+ result.getCode();

System.***out***.println(**"msg="**+ result.getMsg();  
} **else** {  
 System.***out***.println(**"错误消息="**+ result.getMsg());  
}

# 消息中心接口

## 发送消息

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 发送消息 |
| 接口描述 | 发送消息 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| merchantSerialNo | 商户流水号 | String | 50 | M | 不能重复 |
| title | 题目 | String | 100 | M |  |
| summary | 摘要 | String | 500 | M |  |
| type | 类型 | Integer | 1 | M | 1：活动  2：公告 |
| content | 内容 | String | 500 | M |  |
| users | 用户集合 | List |  | M | 每次最多传5000个用户 |
| userNo | 用户编号 | String | 50 | M |  |
| userName | 用户名 | String | 50 | O |  |
| userPhone | 用户手机号 | String | 11 | M | 手机号不是11位，验证不发送 |
| userId | 用户id | String | 50 | M |  |

例子：

MsgMessageSendDto sendDto = **new** MsgMessageSendDto();  
sendDto.setMerchantSerialNo(**"100000000"**);  
sendDto.setContent(**"内容"**);  
sendDto.setTitle(**"题目"**);  
sendDto.setSummary(**"摘要"**);  
sendDto.setType(1);  
sendDto.setContent(**"内容"**);  
List<MsgMessageSendDto.UserDto> users = **new** ArrayList<MsgMessageSendDto.UserDto>();  
MsgMessageSendDto.UserDto userOne = **new** MsgMessageSendDto.UserDto(**"100"**,**"100"**,**"one"**,**"18525852586"**);  
MsgMessageSendDto.UserDto userTwo = **new** MsgMessageSendDto.UserDto(**"200"**,**"100"**,**"two"**,**"18525852587"**);  
MsgMessageSendDto.UserDto userThree = **new** MsgMessageSendDto.UserDto(**"300"**,**"100"**,**"three"**,**"18525852588"**);  
users.add(userOne);  
users.add(userTwo);  
users.add(userThree);  
sendDto.setUsers(users);  
**msgMessageFacadeService**.send(sendDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDto<Boolean> resultDto = msgMessageFacadeService.send(sendDto);

if(resultDto.isSuccess()){

System.out.println("code="+resultDto.getCode());

System.out.println("msg="+resultDto.getMsg());

}else{

System.out.println("错误消息="+resultDto.getMsg());

}

## 未读消息个数

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 未读消息个数 |
| 接口描述 | 未读消息个数 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| userId | 用户id | String | 50 | M | 单个 |

例子：

String userId= **"202"**;  
MessageResultDto<Integer> resultDto = **msgMessageFacadeService**.unReadCount(userId);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |
| data | 返回结果集 | Integer |  | 未读个数 |

例子：

**if**(resultDto.isSuccess()){  
 System.***out***.println(**"code="**+resultDto.getCode());  
 System.***out***.println(**"msg="**+resultDto.getMsg());  
 System.***out***.println(**"count="**+resultDto.getData());  
}**else**{  
 System.***out***.println(**"错误消息="**+resultDto.getMsg());  
}

## 读取消息

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 读取消息 |
| 接口描述 | 设置为已读 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| ids | 消息Id | List<Integer> |  | 0 | 全读可不传 |
| userId | 用户id | String | 50 | M | 全读，跟userNo二传一 |
| userNo | 用户编号 | String | 50 | O | 全读必传 |
| isAllRead | 是否全读 | Boolean |  | 0 | False:不全读  True:全读  默认为False |

例子：

MsgMessageReadDto readDto = **new** MsgMessageReadDto();  
readDto.setUserNo(**"200"**);  
readDto.setIsAllRead(**false**);  
List<Integer> ids = **new** ArrayList<Integer>();  
ids.add(1);  
ids.add(2);  
ids.add(3);  
readDto.setIds(ids);  
MessageResultDto<Boolean> resultDto = **msgMessageFacadeService**.read(readDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

**if**(resultDto.isSuccess()){  
 System.***out***.println(**"code="**+resultDto.getCode());  
 System.***out***.println(**"msg="**+resultDto.getMsg());  
}**else**{  
 System.***out***.println(**"错误消息="**+resultDto.getMsg());  
}

## 获取消息详情内容

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 获取消息详情-列表 |
| 接口描述 | 列表 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| ids | Id集合 | List |  | M |  |
| id | 消息id | Integer | 12 | M | 单个消息Id |

例子：

List<Integer> ids = **new** ArrayList<Integer>();  
ids.add(1);  
ids.add(15);  
ids.add(17);  
MessageResultDto<List<MsgMessageDto>> resultDto = **msgMessageFacadeService**.getByIds(ids);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |
| data | 结果集 |  |  | MsgMessageDto:  id:消息id  merchantSerialNo:商户流水号  userNo:用户编号  userId:用户id  userName:用户名  userPhone:用户手机号  title：题目  summary:摘要  content:内容  type:类型;1:活动；2：公告  Status:状态；1：未读；2：已读  createTime:创建时间  modifyTime:修改时间  operator:操作者 |

例子：

MessageResultDto<List<MsgMessageDto>> resultDto = **msgMessageFacadeService**.getByIds(ids);  
**if**(resultDto.isSuccess()){  
 List<MsgMessageDto> msgMessageDtos = resultDto.getData();  
**for**(MsgMessageDto msgMessageDto : msgMessageDtos){  
 System.***out***.println(**"用户名="**+msgMessageDto.getUserName());  
 System.***out***.println(**"类型="**+msgMessageDto.getType().getValue());  
 System.***out***.println(**"手机号="**+msgMessageDto.getUserPhone());  
 }  
}**else**{  
 System.***out***.println(**"错误消息="**+ resultDto.getMsg());  
}

## 获取消息列表-分页

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 获取消息列表 |
| 接口描述 | 分页 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| pageNo | 第几页 | Integer | 2 | 0 | 默认为1 |
| pageSize | 每页显示条数 | Integer | 2 | 0 | 默认为20 |
| userNo | 用户编号 | String | 50 | O |  |
| userId | 用户id | String | 50 | O |  |
| merchantSerialNo | 商户号 | String | 50 | O |  |
| title | 题目 | String | 100 | O |  |
| summary | 摘要 | String | 500 | O |  |
| content | 内容 | String | 500 | O |  |

例子：

MsgMessageSearchDto searchDto = **new** MsgMessageSearchDto();  
searchDto.setPageNo(2);  
searchDto.setPageSize(4);  
MessagePageResultDto<MsgMessageDto> resultDto = **msgMessageFacadeService**.search(searchDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |
| data | 结果集 |  |  | MsgMessageDto:  id:消息id  merchantSerialNo:商户流水号  userNo:用户编号  userId:用户id  userName:用户名  userPhone:用户手机号  title：题目  summary:摘要  content:内容  type:类型;1:活动；2：公告  Status:状态；1：未读；2：已读  createTime:创建时间  modifyTime:修改时间  operator:操作者 |

例子：

MessagePageResultDto<MsgMessageDto> resultDto = msgMessageFacadeService.search(searchDto);

if(resultDto.isSuccess()){

System.out.println("code="+resultDto.getCode());

System.out.println("msg="+resultDto.getMsg());

List<MsgMessageDto> msgMessageDtos = resultDto.getDataList();

for(MsgMessageDto msgMessageDto : msgMessageDtos){

System.out.println("用户名："+msgMessageDto.getUserName());

System.out.println("内容："+msgMessageDto.getContent());

}

}else{

System.out.println("错误消息="+resultDto.getMsg());

}

# 短信接口

## 发送验证码短信

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 发送验证码短信 |
| 接口描述 | 发送验证码短信 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| mobile | 手机号码 | String | 11 | M | 11位手机号码 |
| verifyCode | 验证码 | String | 10 | M | 验证码 |
| sendMsg | 发送内容 | String | 100 | M | 例如：你的验证码是1234,有效期10分钟 |

例子：

SendVerifyCodeMsgReqDto reqDto = **new SendVerifyCodeMsgReqDto("18102110988", "2566", "你的验证码是1234,有效期10分钟");**MessageResultDto<SendVerifyCodeMsgRspDto> rspDto = **smFacadeService**.sendVerifyCodeMsg(reqDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDto rspDto = smFacadeService.sendVerifyCodeMsg(reqDto);

if(rspDto.isSuccess()){

System.out.println("code="+ rspDto.getCode());

System.out.println("msg="+ rspDto.getMsg());

}else{

System.out.println("错误消息="+ rspDto.getMsg());

}

## 发送通知短信

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 发送通知短信 |
| 接口描述 | 发送通知短信 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| mobile | 手机号码 | String | 11 | M | 11位手机号码 |
| sendMsg | 发送内容 | String | 100 | M | 发送内容 |

例子：

SendSmNotifyReqDto reqDto = **new** SendSmNotifyReqDto(**"18102110988"**, **"这是通知"**);  
MessageResultDto rspDto = **smFacadeService**.sendNotifyMsg(reqDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 返回码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDto rspDto = smFacadeService.sendNotifyMsg(reqDto);

if(rspDto.isSuccess()){

System.out.println("code="+ rspDto.getCode());

System.out.println("msg="+ rspDto.getMsg());

}else{

System.out.println("错误消息="+ rspDto.getMsg());

}

## 发送营销短信

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 发送营销短信 |
| 接口描述 | 发送营销短信 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| batchNo | 商户批次号 | String | 50 | M | 商户请求批次号，不能重复 |
| mobiles | 手机号列表 | List<String> |  | M | 最多1000个号码 |
| sendMsg | 发送短信内容 | String | 500 | M |  |
| isInstant | 是否立即推送 | boolean |  | M | true:是 false:否 |
| sendTime | 发送时间 | Date |  | C |  |

例子：

SendSmReqDto reqDto = **new** SendSmReqDto();  
reqDto.setBatchNo(**"0100120001"**);  
reqDto.setSendMsg(**"营销短信"**);  
reqDto.setInstant(**true**);  
reqDto.setSendTime(DateUtils.*parse*(**"2016-11-14 10:00:00"**));  
List<String> mobiles = **new** ArrayList<>();  
mobiles.add(**"18650309360"**);  
mobiles.add(**"18650309362"**);  
reqDto.setMobiles(mobiles);  
  
MessageResultDto rspDto = **smFacadeService**.sendMarketMsg(reqDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 返回码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDto rspDto = **smFacadeService**.sendMarketMsg(reqDto);  
**if** (rspDto.isSuccess()) {  
 ***log***.info(**"code="** + rspDto.getCode());  
 ***log***.info(**"msg="** + rspDto.getMsg());  
} **else** {  
 ***log***.info(**"错误消息="** + rspDto.getMsg());  
}

## 验证码验证

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 验证码验证 |
| 接口描述 | 验证码验证 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| mobile | 手机号码 | String | 11 | M | 11位手机号码 |
| verifyCode | 验证码 | String | 10 | M |  |

例子：

VerifyCodeReqDto reqDto = **new** VerifyCodeReqDto(**"18102110988"**, **"2566"**);  
MessageResultDto rspDto = **smFacadeService**.verificationCode(reqDto);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDto rspDto = **smFacadeService**.verificationCode(reqDto);  
**if** (rspDto.isSuccess()) {  
 ***log***.info(**"code="** + rspDto.getCode());  
 ***log***.info(**"msg="** + rspDto.getMsg());  
} **else** {  
 ***log***.info(**"错误消息="** + rspDto.getMsg());  
}

# 访问日志接口

## 记录访问日志

### 接口描述

|  |  |
| --- | --- |
| 接口名称 | 记录访问日志 |
| 接口描述 | 记录访问日志 |
| 接口提供者 | 消息中心系统 |
| 接口使用者 | 业务系统 |
| 接口类型 | hessian |
| 应用场景 |  |
| 备注 |  |

### 请求参数

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 必填 | 备注 |
| systemName | 系统名称 | String | 100 | M | 系统名称 |
| methodName | 方法名称 | String | 100 | M | 方法 |
| startTime | 开始时间 | Date |  | M |  |
| endTime | 结束时间 | Date |  | M |  |
| accessMsg | 要记录的信息 | String | 2000 |  |  |

例子：

Date endTime = **new** Date();  
Date startTime = DateUtils.*plusSeconds*(endTime, -14);  
  
LogDataReqDto reqDto = **new** LogDataReqDto(**"Laocaibao"**, **"webInit"**, startTime, endTime，null);

### 请求结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 含义 | 类型 | 长度 | 备注 |
| code | 成功或者失败编码 | string |  | 成功编码:0000  失败编码:1111 |
| msg | 提示信息 | string |  | 失败时候提示信息 |

例子：

MessageResultDto<LogDataRspDto> resultDto = **logFacadeService**.accessLog(reqDto);  
*assertTrue*(resultDto.isSuccess());