

智能客服

API 文档

产品文档



腾讯云

【版权声明】

©2013-2019 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

文档目录

API 文档

问答

敏感词过滤

模版消息推送

API 文档

问答

最近更新时间：2018-11-19 21:50:14

1. 接口描述

此接口用于智能回复用户提问。

调用协议：HTTP/HTTPS GET 请求。

2. 请求参数

参数名称	参数类型	是否必填	参数说明
appid	String	是	应用 ID，可在 IASK 应用管理界面获取
uuld	String	是	用户唯一 ID，最大长度 20 位
channel	Int	是	接口调用渠道，0：问答 API，1：公众号，2：桌面网站，3：移动网站
query	String	是	用户查询问题
time	Int	是	时间戳，单位：秒
sign	String	是	调用签名，根据接口调用参数及应用 SecretKey 生成

3. 响应参数

参数名	类型	说明
code	Int	code 码，0 为正常，非 0 为异常
data	Object	返回结果
msg	String	返回消息

4. 接入流程

4.1 用户从 IASK 的应用获取 APPID 及 SecretKey ；

4.2 确认除 sign 参数外各参数的值 ；

4.3 生成 sign 参数，将 APPID、UUID、channel、query、time 参数按升序排列组合成字符串得到 `&appId=x&channel=x&query=x&time=x&uuld=x` ；在字符串头部加上 SecretKey 得到 `x&appId=x&channel=x&query=x&time=x&uuld=x` 。对此字符串做 MD5 处理得到签名参数 sign。

sign 参数生成示例（PHP）：

```
$secret_key = xxx; //IASK 应用配置里面的 SecretKey
'appld' => 'xx'
'time' => 1525339871,
'channel' => 'xx',
'uuld' => 'xx',
'query' => 'xx',
];
ksort($params);
foreach ($params as $key => $value) {
    $secret_key.= '&' . $key . '=' . $value;
}
$sign = md5($secret_key);
```

5. 示例

请求示例

```
iask.qq.com/aics/open/ask?
appId=817c0cc4fb65b98022b73b56485a515d
&uuld=aa
&channel=1
&query=再来试试
&time=1528012017
&sign=dc0900e8209f1b3c16f4c0bb736d7a39
```

响应示例

直接返回结果：

```
{
    code: 0,
```

```
data: {
  type => 'STRING', //答案结构类型, ( 'STRING' ,' ARRAY' )
  prefix => "", //IASK 提供的前置话术
  answer => 'xxx', //IASK 返回的答案 ( type 为 STRING )
  rel => [
    "xxx",
    "xxx"
  ], //IASK 提供的关联问题集合
  suffix => "", //IASK 提供的后置话术
},
msg: "success"
}
```

追问返回结果：

```
{
  code: 0,
  data: {
    type => 'ARRAY', //答案结构类型, ( 'STRING' ,' ARRAY' )
    prefix => "", //IASK 提供的前置话术
    answer => [
      "xxx",
      "xxx",
    ], //IASK 返回的追问回答集合 ( type 为 ARRAY )
    rel => [], //IASK 提供的关联问题集合
    suffix => "", //IASK 提供的后置话术
  },
  msg: "success"
}
```

敏感词过滤

最近更新时间：2018-11-19 21:50:10

1. 接口描述

此接口用于敏感词过滤。用户咨询输入的内容包含敏感词，该词会用 ** 号展示替代，可在 IASK 管理台的会话设置页面添加敏感词。

调用协议：HTTP/HTTPS GET 请求

2. 请求参数

参数名称	参数类型	是否必填	参数说明
appld	String	是	应用 ID，可在 IASK 应用管理界面获取
query	String	是	待过滤文本
time	Int	是	时间戳，单位：秒
sign	String	是	调用签名，根据接口调用参数及应用 SecretKey 生成

3. 响应参数

参数名称	参数类型	参数说明
code	Int	code 码，0 为正常，非 0 为异常
data	String	返回过滤后的文本
msg	String	返回消息

4. 接入流程

4.1 用户从 IASK 的应用获取 APPID 及 SecretKey；

4.2 确认除 sign 参数外各参数的值；

4.3 生成 sign 参数，将 appld、uuld、channel、query、time 参数按升序排列组合成字符串得

到 `&appId=x&query=x&time=x` ; 在字符串头部加上 `SecretKey` 得到 `x&appId=x&query=x&time=x` 。对此字符串做 MD5 处理得到签名参数 `sign`。

sign 参数生成示例 (PHP) :

```
$secret_key = xxx; //IASK 应用配置里面的 SecretKey
$params = [
    'appId' => 'xx'
    'time' => 1525339871,
    'query' => 'xx',
];
ksort($params);
foreach ($params as $key => $value) {
    $secret_key.= '&' . $key . '=' . $value;
}
$sign = md5($secret_key);
```

5. 示例

请求示例

```
iask.qq.com /aics/open/filterSensitiveWord?
appId=817c0cc4fb65b98022b73b56485a515d
&query=你能不能酸萝卜别吃了啊
&time=1528803562
&sign=a96c71b1d0f192906310bbaccf8ce1d1
```

响应示例

```
{
    "code": 0,
    "data": "你能不能*****了啊", //IASK 过滤后结果, 敏感词会逐字符替换为*
    "msg": "success"
}
```

模版消息推送

最近更新时间：2019-04-25 18:20:02

1. 接口描述

接口请求域名：`iask.qq.com`

本接口（`pushTemplateMsg`）用于推送模板消息。调用协议为 HTTPS POST 请求，`content-type` 为 `application/JSON`。

2. 请求参数

参数名称	参数类型	是否必填	参数说明
<code>appId</code>	String	是	应用 ID，可在 IASK 应用管理界面获取
<code>oaid</code>	String	是	应用授权给 IASK 的公众号 ID
<code>openId</code>	Int	是	接收者 openid
<code>templateId</code>	String	是	模板 ID
<code>url</code>	String	否	模板跳转链接
<code>timestamp</code>	Int	是	时间戳，单位秒
<code>sign</code>	String	是	调用签名，根据接口调用参数及应用 SecretKey 生成
<code>miniprogram</code>	JSON	否	放置在 post body 中，详见 示例1
<code>data</code>	JSON	是	放置在 post body 中，详见 示例2

3. 响应参数

参数名称	参数类型	参数说明
<code>code</code>	Int	code 码，0为正常，非0为异常
<code>data</code>	String	返回过滤后的文本

参数名称	参数类型	参数说明
msg	String	返回消息

4. 接入流程

4.1 用户从 IASK 的应用获取 APPID 及 SecretKey ;

4.2 确认除 sign 参数外各参数的值 (不需要考虑 postbody 里面的参数) ;

4.3 生成 sign 参数, 将 APPID、oalD、openID、templateID、timestamp 参数按升序排列组合成字符串得到 `&appld=x&openId=x&templateId=x×tamp=x` ; 在字符串头部加上 SecretKey 得到 `x&appld=x&openId=x&templateId=x×tamp=x` 。对此字符串做 MD5处理得到签名参数 sign。

sign 参数生成示例 (PHP) :

```
$secret_key = xxx; //IASK 应用配置里面的 SecretKey
$params = [
    'appld' => 'xx'
    'openId' => 'xx'
    'timestamp' => 1525339871,
    'oalD' => 'xx',
    'templateId' => 'xx',
];
ksort($params);
foreach ($params as $key => $value) {
    $secret_key.= '&' . $key . '=' . $value;
}
$sign = md5($secret_key);
```

5. 示例

请求示例

```
iask.qq.com/aics/open/pushTemplateMsg?
appld=be043fdb5e7d23041f0436a27ae3777d
&oalD=wxc300f0df2a16da43
&openId=ogud103JWOxWQmSKy2DR-NE6NTD4
&templateId=Jpdhk-mCTam7fQjbObwBgZW54g2ij8cja7NYo_b8YU
&url=http://www.qq.com&timestamp=111&sign=xxx
```

示例1 (miniprogram) :

```
{
  "appid": "xiaochengxuappid12345", //小程序 APPID
  "pagepath": "index?foo=bar" //小程序具体页面地址
}
```

示例2 (data) :

```
{
  "first": {
    "value": "工单处理进度通知",
    "color": "#173177"
  },
  "keyword1": {
    "value": "xxx",
    "color": "#173177"
  },
  "keyword2": {
    "value": "xxx",
    "color": "#173177"
  },
  "keyword3": {
    "value": "xxx",
    "color": "#173177"
  },
  "remark": {
    "value": "请点击查看详情!",
    "color": "#173177"
  }
}
```

响应示例

```
{
  "code": 0,
  "data": {
    "errcode": 0, //推送模板消息接口返回 code 码
    "errmsg": "ok", //推送模板消息接口返回 msg
    "msgid": 496146255857205248 //推送模板消息接口返回 msgId
  },
  "msg": "success"
}
```