

1. 目的:  
用于将机构网站或者落地页的客户数据发送到云朵 crm 中.
2. 版本说明:
  - (1) 请求的路径为 `xx/externalData/standardClueAccept/xx` 的为 v1 版本接口;
  - (2) 请求的路径 `xx/external/clue/standardClueAccept/v2/xx` 的为 v2 版本接口
  - (3) v1 接口和 v2 接口的请求方式不同,v1 为历史配置兼容旧的请求方式,v2 为新的配置之后的创建配置都使用此请求方式.
3. v2 版本请求说明及 demo (新建的表单一定是 v2 的或请求路径里有 v2 标记的)
  - (1) 请求说明
    - ① 请求方式 :POST
    - ② contentType :application/json
    - ③ 多值数据或级联数据 传英文逗号分隔的字符串
  - (2) 参数  
假如配置方式为:

[+添加映射字段](#)

crm字段	数据类型	字段名称	操作
手机号码	输入框	phone	删除
客户姓名	输入框	name	删除

那么请求格式为

```
{
  "phone":"18888888888",
  "name":"测试 001",
  "ip":"123.118.99.19",
  "url":"http://test.site.com"
}
```

说明:phone 和 name 根据页面配置项传输,ip 和 url 为固定参数,非必填

ip 会直接赋值到客户字段 - 访问 ip 如果不传,就获取请求 ip

url 为落地页 url,会直接赋值到客户字段- 落地页链接,如果是从页面直接的请求,会自动获取;  
鉴权方式

请求的 header 中必传 3 个参数 timeStamp ,token ,sign

timeStamp 当前时间的 unix 时间戳(可获取当前时间的毫秒数 除以 1000)

token 从页面中拷贝的 token

sign 将参数根据 key 排序拼接成字符串,加 token 加 时间戳 ,使用 MD5 进行加密处理的结果值;

请求示例:

- ① Java

```
import com.alibaba.fastjson.JSON;
import com.yx.crm.util.MD5;
```

```

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.HashMap;
import java.util.Map;
import java.util.Set;

public class RequestDemo {

    public static void main(String[] args) {

        //页面拷贝
        String url = "";

        //页面拷贝
        String token = "";

        String timeStamp = System.currentTimeMillis() / 1000 + "";

        //根据页面中的映射配置
        Map<String, Object> params = new HashMap<>();

        params.put("tel", "16666666666");

        params.put("name", "测试名称");

        String paramSortStr = paramSortStr(params);

        System.out.println(paramSortStr);

        Map<String, String> heads = new HashMap<>();

        heads.put("timeStamp", timeStamp);

        heads.put("token", token);

        heads.put("sign", createStandardDataSign(timeStamp, token, params));

        String result = sendPostJson(url, JSON.toJSONString(params), heads);

        System.out.println(result);

    }

    /**
     * 生成通用数据接口的签名
     * @return
     */

    public static String createStandardDataSign(String timeStamp, String token, Map<String, Object> reqParam) {

```

```

String paramSortStr = paramSortStr(reqParam);

return MD5.getMD5(paramSortStr + token + timeStamp);
}

public static <K extends Comparable<? super K>, V> String paramSortStr(Map<K, V> map) {

    StringBuilder paramStr = new StringBuilder(map.size() * 8);

    map.entrySet().stream().filter(x -> x.getValue() != null)

        .sorted(Map.Entry.<K, V>comparingByKey()

            ).forEachOrdered(e -> paramStr.append(e.getKey()).append("=").append(e.getValue().toString()).append("&"));

    paramStr.deleteCharAt(paramStr.length() - 1);

    return paramStr.toString();
}

private static String sendPostJson(String url, String jsonStr, Map<String, String> headers) {

    PrintWriter out = null;

    BufferedReader in = null;

    String result = "";

    try {

        URL urlObj = new URL(url);

        HttpURLConnection conn = (HttpURLConnection) urlObj.openConnection();

        conn.setRequestMethod("POST");

        conn.setRequestProperty("accept", "*/*");

        conn.setRequestProperty("Content-Type", "application/json;charset=utf-8");

        conn.setRequestProperty("connection", "Keep-Alive");

        if (headers != null && headers.size() > 0) {

            Set<Map.Entry<String, String>> headerEntries = headers.entrySet();

            for (Map.Entry<String, String> headerEntry : headerEntries) {

                conn.setRequestProperty(headerEntry.getKey(), headerEntry.getValue());

            }

        }

        conn.setDoOutput(true);

        conn.setDoInput(true);

        out = new PrintWriter(conn.getOutputStream());

        out.write(jsonStr);

        out.flush();

        in = new BufferedReader(new InputStreamReader(conn.getInputStream()));

        String line;

        while ((line = in.readLine()) != null) {

            result += line;

        }

    } catch (IOException e) {

        e.printStackTrace();
    }
}

```

```
    } finally {

        try {

            if (out != null) {

                out.close();

            }

            if (in != null) {

                in.close();

            }

        } catch (IOException ex) {

            ex.printStackTrace();

        }

    }

    return result;

}

}
```

② js

```
<script type="text/javascript" src="https://crm.yunduocrm.com/resources/plugins/jquery/jquery-1.9.1.min.js" ></script>

<script type="text/javascript" src="https://crm.yunduocrm.com/resources/js/md5.js" ></script>

<script type="text/javascript">

    var reqUrl = "";

    var token = "";

    var timeStamp = ((new Date()).getTime() / 1000) | 0 + "";

    var reqParm = {

        "tel": "16666666668",

        "name": "测试名称"

    }

    var sign = $.md5(sortKeyStr(reqParm) + token + timeStamp);

    function sortKeyStr(obj){

    var str = "";

    Object.keys(obj).sort().forEach(function(key) {

        str += key + "=" + obj[key] + "&";

    });

    return str.substr(0, str.length - 1);

}
```

```

    }

    $.ajax({
        type:"post",
        data:JSON.stringify(reqParm),
        headers:{
            "timeStamp":timeStamp,
            "token":token,
            "sign":sign,
        },
        url:reqUrl,
        dataType: "json",
        contentType:"application/json;charset=utf-8",
        success: function(result){
            console.log(result);
        }
    });
</script>

```

- (3) 返回参数说明:  
成功响应 {"code":200}

#### 4. V1 版本请求方式说明 (旧配置项)

##### (1) 请求

请求方:POST

Contenttype:application/x-www-form-urlencoded (注意非 json 的方式请求)

参数名称	类型	是否必须	说明
token	string	是	页面复制
data_body	string	是	将数据拼成 json 字符串后,作为此参数的值 格式 [ <pre> {"name":"tel","value":"15844747458"}, {"name":"name","value":"名称"}                     </pre> ]
data_url	string	否	数据的落地页 url

##### (2) 参数说明

### 若为此配置项

CRM 字段	映射方式	API 字段名
行政地区	多级下拉框	area
客户来源	下拉菜单	source
手机号码	电话	tel
客户姓名	输入框	name

配置传入的参数与 crm 中字段的映射关系,右侧输入框为传入的字段 name,对应的 data\_body 数据

```
{
  "name": "area",
  "value": "北京市,北京市,朝阳区"
}, {
  "name": "source",
  "value": "官网咨询"
}, {
  "name": "tel",
  "value": "17633226563"
}, {
  "name": "name",
  "value": "张三"
}
```

### (3) 请求示例

POST [https://crm.yunduocrm.com/externalData/standardClueAccept/1886/531075ae4cc84c10af14a84a1e8afbe5?router\\_id=1886](https://crm.yunduocrm.com/externalData/standardClueAccept/1886/531075ae4cc84c10af14a84a1e8afbe5?router_id=1886)

Params Authorization Headers (9) **Body** Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE
<input checked="" type="checkbox"/> token	ZjM4NjJkMGZlY2Q3YzlhNzYxZGI0M2YyZDZlY2ExYTdhMzJkZlZlMjY3ZmF1MWFkODQxYzZmMz...
<input checked="" type="checkbox"/> data_body	[   ... {"name":"tel","value":"15844747458"},   ... {"name":"name","value":"李四"},   ... {"name":"source","value":"自主报名"}, ]
Key	