目录

[游戏统计SDK服务端文档 1](#_Toc403059793)

[一、 功能： 1](#_Toc403059794)

[二、接口 1](#_Toc403059795)

[获取web后台配置的信息(自定义事件等) 1](#_Toc403059796)

[上传应用的统计信息。 2](#_Toc403059797)

游戏统计SDK服务端文档 （第一版）

# 功能：

1. Web管理后台配置自定义事件(eventId：eventVal)。
2. 统计：
   1. 每个页面显示的时长
   2. 自定义事件
   3. 程序异常捕获
   4. 程序启动时间
   5. 渠道分发 (新增)
   6. 关卡 (新增)
   7. 消费 (新增)
   8. 用户信息（新增）

# 二、接口

## 获取web后台配置的信息(自定义事件等)

地址：[www.xxx.com/getOnlineConfig](http://www.xxx.com/getOnlineConfig)

参数：mf\_appkey :每个应用对应的唯一key，由服务端生成。

返回数据格式：json

{

events:[

{"事件1":"值1"}

{"事件2":"值2"}

],

goods:[

{“goodId1”,”good1”},

{“goodId2”,”good2”}

],

levels:[

{"levelId1":"level1"}

{" levelId2":" level2"}

],

}

## 上传应用的统计信息。

用户在第二次启动应用时会上传上次启动的统计信息，假如没有网络，则会暂时将统计信息缓存到本地。

地址：[www.xxx.com/submitAnalyticInfo](http://www.xxx.com/submitAnalyticInfo)

参数：mf\_appkey :每个应用对应的唯一key，由服务端生成。

phoneId:根据用户手机硬件信息生成的唯一标识符.

Info:统计信息，格式：json。示例：

{

//launch:启动信息

“launch”：**[**

{

“date”: "2014-10-31",//启动日期

“session\_id”: "74E7D9DDC190A146861CE91ADEB9B3F8",

//每次启动生成的UUID，当应用两次使用间隔

//大于30s，则认为是两次启动。

“time”: "14:31:22"//启动时间

} ,

{

“date”: "2014-11-1",

“session\_id”: "B0D4EA8E30B90FE9C5B9D63E2C8D1AC0",

“time”: "11:11:11"

}

**],**

//events:自定义事件，只有在web后台定义了才会被统计。

“events”:**[**

//第一种事件：只有eventId

{

“session\_id“: "78446C6849B461A5F89B37CB203C6775",

“time“: "14:18:04",

“date“: "2014-10-31",

“eventId“: "clickEvent"

},

//第二种事件：有eventId和一个属性的值

{

“session\_id“: "78446C6849B461A5F89B37CB203C6775",

“time“: "14:18:04",

“label“: "btn",//属性的值

“date“: "2014-10-31",

“eventId“: " clickEvent "

},

//第三种事件：类似第一种事件，多了 事件时长

{

“dur“:132, //**单位：毫秒**

“session\_id“: "78446C6849B461A5F89B37CB203C6775",

“time“: "14:18:04",

“date“: "2014-10-31",

“eventId“: "clickEvent"

},

//第四种事件：类似第二种事件，多了 事件时长

{

“dur“:666, //**单位：毫秒**

“session\_id“: "78446C6849B461A5F89B37CB203C6775",

“time“: "14:18:04",

“label“: "btn",//属性的值

“date“: "2014-10-31",

“eventId“: "clickEvent"

},

**],**

//ekv:另一类自定义事件，上面几种事件的属性值要么没有，要么有一个， 如果事件有多个属性值，需要用这种自定义事件。

“ekv”:**[**

//第一种

{

“session\_id“: "78446C6849B461A5F89B37CB203C6775",

“time“: "14:18:04",

“date“: "2014-10-31",

“eventId“: "clickEvent",

“key1”:”value1”,//属性1

“key2”:”value2”//属性2

},

//第二种，有 事件时长

{

“dur”:333,//**单位：毫秒**

“session\_id“: "78446C6849B461A5F89B37CB203C6775",

“time“: "14:18:04",

“date“: "2014-10-31",

“eventId“: "clickEvent",

“key1”:”value1”,//属性1

“key2”:”value2”//属性2

}

**]**

//统计每个页面显示的时长

“terminate”:[

//第一次启动

{

duration: 24,//本次启动所有页面显示总时长 单位：秒

session\_id: "B0D4EA8E30B90FE9C5B9D63E2C8D1AC0",

time: "14:30:51",

date: "2014-10-31",

pages: [

[

"AnalyticsHome",”11” //这个页面显示了11秒

],

[

"AnalyticsHome",”3” //这个页面又显示了3秒

],

[

"FragmentSimple",2

],

[

"FragmentContacts",0

]

]

},

//第二次启动

{

//省略

}

]

//程序异常信息

“errors”:[

{“content”:”java.lang.exception at xxxx”

},

{“content”:”java.lang.exception at xxxx”

}

]

//游戏统计 包括:⑴关卡:关卡开始/失败/完成 ⑵消费:真实消费(花RMB的))/虚拟消费 ⑶

“gkv”:[

{

id: "level",

user\_level: "LV.01",//用户角色的级别

level: "level-1",//关卡id,同eventId一样,web后台配置后才会被统计

ts: 1415002272,//当前时间,计算方式:系统时间/1000 单位:秒

status: 0, //关卡状态: 0代表开始 1代表完成 -1代表失败

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

{

id: "level",

user\_level: "LV.01",

duration: 14981,//从关卡开始到关卡失败的时长 单位:毫秒

level: "level-1",

ts: 1415002287,

status: -1, //关卡失败

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

{

id: "level",

user\_level: "LV.01",

level: "level-1",

ts: 1415002288,

status: 0,

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

{

id: "level",

user\_level: "LV.01",

duration: 1017,

level: "level-1",

ts: 1415002289,

status: 1, //关卡完成

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

//第一种真实消费:花10元购买2000金币

{

cash: 1000,//人民币 单位:分

id: "pay",//真实消费用pay 虚拟消费用buy

user\_level: "LV.01",

level: "level-1",

source: 2,//支付渠道 详见下表

ts: 1415002291,

coin: 2000,//虚拟货币 单位:个

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

//第二种真实消费:花10元直接购买某个道具,价值1000金币

//此种消费分为以下两个json数据,第一个是花10元购买1000金币,第二个是1000金币购买某个道具

{

"cash": 1000, //人民币 单位:分

"id": "pay",//真实消费用pay 虚拟消费用buy

"user\_level": "LV.01",

"level": "level-1",

"source": 2,

"ts": 1415244519,

"coin": 1000,

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

{

“appkey” : “”,

“phoneid” : “”

"cash": 1000, //人民币 单位:分

"id": "pay",//真实消费用pay 虚拟消费用buy

"user\_level": "LV.01",

"level": "level-1",

"source": 2,

"ts": 1415244519,

"coin": 1000,

session\_id: "7392EC547A7A0FD493B39444DA896759"

}

{

"amount": 1,

"id": "buy",

"user\_level": "LV.01",

"level": "level-1",

"ts": 1415244519,

"coin": 1000,

"item": "sword",

session\_id: "7392EC547A7A0FD493B39444DA896759"

}

//虚拟消费 用虚拟币购买物品

{

amount: 2,//购买数量

id: "buy",

user\_level: "LV.01",

level: "level-1",

ts: 1415002293,

coin: 200,//消费的虚拟货币 单位:个

item: "magic",//物品名称

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

//物品消耗

{

amount: 1,

id: "use” //使用物品用use

user\_level: "LV.01",

level: "level-1",

ts: 1415002292,

coin: 1000,//消费的虚拟货币 单位:个

item: "sword"//使用的物品名称

},

//奖励 第一种: 奖励虚拟货币

{

"id": "bonus",//奖励的id是 bonus

"user\_level": "LV.01",

"bonus\_source": 2,//奖励来源取值在 1~10 之间，“1”已经被预先定义为“系统奖励”， 2~10 需要在网站设置含义。

"level": "level-1",

"ts": 1415006963,

"coin": 1000,//奖励1000金币

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

//奖励 第二种:奖励物品. 每次奖励事件都有两个json对象,第一个是用户购买物品 , 第二个是奖励物品

{

"amount": 1,

"id": "buy",

"user\_level": "LV.01",

"level": "level-1",

"ts": 1415006963,

"coin": 100,

"item": "sword",

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

{

"id": "bonus",

"user\_level": "LV.01",

"bonus\_source": 1,

"level": "level-1",

"ts": 1415006963,

"coin": 100,

session\_id: "7392EC547A7A0FD493B39444DA896759"

},

]

}

**支付渠道表:**

| **数值** | **含义** |
| --- | --- |
| 1 | App Store |
| 2 | 支付宝 |
| 3 | 网银 |
| 4 | 财付通 |
| 5 | 移动通信 |
| 6 | 联通通信 |
| 7 | 电信通信 |
| 8 | paypal |

这8种是预定义的,系统预留了1- 20,用户可以在web后台自定义渠道,数值从21开始.

1. 玩家信息统计

地址：www.xxx.com/userInfo

参数：mf\_appkey :每个应用对应的唯一key，由服务端生成。

userInfo : 格式:json

{

Id: 玩家ID   
age: 玩家年龄   
gender: 玩家性别,0代表男 1代表女 2代表未知  
source: 玩家Id来源标示，比如”微信”，”微博”等

}

1. 渠道分发 获取应用是从哪个渠道(360/豌豆荚)等下载的.

地址：[www.xxx.com/channelInfo](http://www.xxx.com/channelInfo)

参数: mf\_appkey :每个应用对应的唯一key，由服务端生成。

phoneId:根据用户手机硬件信息生成的唯一标识符.

meta\_data:”360”