1. 获取多车的日志类型

请求json：

{

event: get\_log\_types

login\_id:1

robot\_id: [id0,id1,id2](int)

}

反馈json

{

code:1 =0 :成功,!=0 失败,

msg: 描述

data: [type1,type2,type3,.....]

}

1. 从车上获取日志

请求json：

{

event: send\_log\_condition

login\_id:1

robot\_id: [id0,id1,id2]

start\_time:20180601\_103000

%Y%M%d\_%H%m%s

end\_time:20180630\_123000

is\_latest\_time:0 (0:非最新时间，1:使用最新时间)

time\_select\_type:0 (0:昨天，1:今天)

type\_list:[type0,type1,type2...]

name : log1

}

反馈json

{

code:1 =0 :成功,!=0 失败,

30X:时间不合法

msg: 描述

task\_id:1

}

1. 取消拉日志任务

请求json:

{

event:cancle\_get\_log

login\_id:1

task\_id:1

}

反馈json

{

code:1 =0 :成功,

30X:任务不存在

msg: 描述

result:0 (>=0 :成功,<0 失败)

}

1. 进度反馈

sockit io

{

msg\_type:0x00000106

user\_id:1

task\_id:1

step:99

}

1. 正在上传的任务

请求json:

{

event:get\_executing\_log

login\_id:1

}

反馈json

{

code:1 =0 :成功,!=0 失败,

30X+1:任务已存在

msg: 描述

log\_name:2.tar

task\_id:1（-1：表示无任务）

step:88（-1：无任务）

}

1. 删除后台文件

请求json:

{

event:delete\_log

login\_id:1

log\_name:’111.tar’

}

反馈json

{

code:1 =0 :成功,!=0 失败,

系统错误

msg: 描述

}

1. 后台存在的文件列表

请求json:

{

event:get\_log\_list

login\_id:1

}

反馈json

{

code:1 =0 :成功

数据库底层数据错误

msg: 描述

log\_list:[{‘path’:log1.tar,’size’:12B,’time’:},{}]

}

1. ~~后台文件下载到用户本地~~

~~请求json:~~

~~{~~

~~event:download\_log~~

~~login\_id:1~~

~~log\_name:’111.tar’~~

~~}~~

~~反馈json~~

~~{~~

~~code:1 =0 :成功,!=0 失败,~~

~~msg: 描述~~

~~path:/website/user/blackbox/2.tar~~

~~}~~

二、备份配置文件

1. ~~1、删除后台文件~~

~~请求json:~~

~~{~~

~~event:backup\_config\_file~~

~~user\_id:1~~

~~robot\_id: [id0,id1,id2](int)~~

~~}~~

~~反馈json~~

~~{~~

~~code:1 =0 :成功,!=0 失败,~~

~~msg: 描述~~

~~}~~

归类到用户错误区间中

**1. 插入模板**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 类型 | 必选 | 备注 |
| event | 事件类型 | string | 是 | event\_bk\_temps\_insert |
| login\_id |  | uint32 |  |  |
| name | 模板名 | string | 是 |  |
| logger\_types | 日志类型 | string | 是 |  |
| others | 其他类型 | string | 是 |  |

请求json实例：

{

event: " event\_bk\_temps\_insert "

login\_id:1

name: “123”

logger\_types: “”

others: “”

}

反馈json

{

code:0:成功,非0失败

}

**2.删除模板**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 类型 | 必选 | 备注 |
| event | 事件类型 | string | 是 | event\_bk\_temps\_remove |
| login\_id |  | uint32 |  |  |
| temps\_id | 模板id | int | 是 |  |

请求json实例：

{

event: " event\_bk\_temps\_remove "

login\_id:1

temps\_id: 1

}

反馈json

{

code:0:成功,非0失败

}

**3.模板列表**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数名 | 含义 | 类型 | 必选 | 备注 |
| event | 事件类型 | string | 是 | event\_bk\_temp\_list |
| login\_id | 用户id | int | 是 |  |

请求json实例：

{

event: " event\_bk\_temp\_list "

login\_id: 1

}

反馈json

{

code:0:成功,非0失败

data：{

temps:[{‘id’:1,

‘name’: 模板名称

‘logger\_types：日志列表

‘others’： 其他日志列表

‘time’：时间

}]

}

}