

米果智能_晨检仪 APP-对接 第三方业务后台 API-1.0

文档说明

● 版本说明

2024-12:

1.初版;

● 阅读对象

本文阅读对象：第三方对接晨检仪 APP 的开发工程师;参与果智能开放平台的技术架构师，研发工程师，测试工程师，系统运维工程师。

● 接入规范

接入 API 约定以下几项规范：

通信方式	采用 HTTP 协议，以 POST/GET 方式提交请求
数据格式	采用 JSON 格式
字符编码	采用 utf-8
签名算法	暂支持 MD5

● 安全规范（签名算法）**暂不开启**

1. 设所有发送或者接收到的参数名 ASCII 码从小到大排序（字典序），使用 URL 键值对的格式（即 key1=value1&key2=value2...）最后拼接成字符串 stringA，

规则如下：参数名 ASCII 码从小到大排序（字典序）；如果参数的值为空不参与签名；参数名区分大小写；sign 参数不参与签名；


2. 在 stringA 最后拼接上 key（**米果分配，注意保密**）得到 stringSignTemp 字符串，并对 stringSignTemp 进行 MD5 运算，再将得到的字符串所有字符转换为小写，得到 sign 值 signValue。

补充说明

- 1.下面常提到的 equipmentCode 为设备 ID,每个设备的唯一标识字符串;在应用后台的设备 ID 菜单获取;

2. 服务端返回必须为 json 字符串;

Content-Type: text/html;charset=UTF-8

▼ Response Headers <input type="checkbox"/> Raw	
Connection:	keep-alive
Content-Encoding:	gzip
Content-Type:	text/html;charset=UTF-8
Date:	Thu, 30 Nov 2023 06:54:20 GMT
Keep-Alive:	timeout=4
Proxy-Connection:	keep-alive
Server:	nginx/1.10.2
Transfer-Encoding:	chunked 
Vary:	Accept-Encoding

API 列表

1. 员工列表 (主流程 API)

示例：https://docs.apipost.net/docs/detail/4efd24986418000?target_id=2fd244c13b0014

接口链接: **\$(第三地服务域名或 ip 地址)/device/morningChecker/employeeList**

请求方法: GET

说明:

请求参数:

变量名	字段名	是否必传	类型	示例值	描述
equipmentCode	设备 ID	是	String	6ac84eb75dcf44dc	设备的唯一 ID[设备后台管理点设备 ID 获取]

提交数据范例如下：

```
1  {
2    "equipmentCode":"ac00146471d2c711",
3  }
4
```

返回结果

变量名	字段名	是否必传	类型	示例值	描述
statusCode	请求状态码	是	int	200/300	200 为操作成功,其他为操作失败;
message	返回信息	是	String	"操作成功"	成功提示的信息 失败提示的信息;
data	员工数据列表	是	JsonArray	如下图所示	

返回结果格式如下：

1.健康证到期日期是 healthEndTime；如果要显示，那么 healthStartTime,healthEndTime 要同时都有数据

```
1  {
2    "data": [ // 员工列表数据
3
4      {
5        "healthEndTime": "2026-11-05", //健康证结束日期
6        "healthStartTime": "2024-12-03", // 健康证开始日期
7        "healthNumber": "2222", // 健康证号码
8        "healthUrl":
9          "http://picture-mgcr.megoai.com/lw/1/healthUrl/8ad68484-b27a-43b1-837c-2650
10         306a95e8.jpg", // 健康证图片查看链接
11        "name": "李食神", // 名字名称 (不可为空)
12        "post": "总厨", // 职位,岗位
13        "portraitPhoto":
14          "http://picture-mgcr.megoai.com/lw/1/datadrive/mego/face/helper/1/126.jpg?v=
15          1734571595661", // 用户头像图片链接 (注意:有且仅有只能出现一个人脸,保证相片不模
16          糊和不要逆光,否则人脸将注册不成功)
17
18        "userId": "126", // 用户 ID 字符串类型 (不可为空)
19        "updateTime": "2024-12-25 12:00:00" // 员工信息修改时间
20      },
21      ...
22    ],
23    "message": "成功",
24    "statusCode": 200
25  }
```

2. 提交晨检数据 (主流程 API)

示例：

https://docs.apipost.net/docs/detail/3beb4c9f8403000?locale=zh-cn&target_id=31d41beff63000

接口链接：

`${第三地服务域名或 ip 地址}/device/morningChecker/checkData`

请求方法: **POST**

请求参数:

变量名	字段名	是否 必传	类型	示例值	描述
machineCode	设备 ID	是	String	6ac84eb75dcf44dc	设备的唯一 ID

提交数据范例如下：

formdata

```
1
2  {
3      'equipmentCode': "asdsdsdsa2342", // 设备 ID
4      'uuid': "1232312dsdsfdsg", // 唯一标识 ID
5      'userId': "123", // 用户 ID 字符串类型(对应员工列表 API 的 userId),
6      'foreheadTemp': 36.6, // 温度, 浮点类型
7      'checkTime': "2024-12-20 18:00:00", // 晨检时间
8      'abnormalTemp': "0", // 温度是否异常 0 正常 1 异常 基于下面的
normalTemperatureRange 范围判断, 温度异常的逻辑为: foreheadTemp > 37.3 ||
foreheadTemp < 35.9
9      'normalTemperatureRange': "35.9-37.3", // 正常温度范围 左边为正常温度范围的最
最小值,右边为正常温度的最大值,
10      // 手部检测情况 无明显异常结果时为空字符串, 其余结果为 异物 创可贴 戒指 指甲
油 伤口 灰指甲 (多个用,号分开)
11      "handCheckResult": "创可贴,指甲油", // 手部检测结果
12      // 健康询问 健康为空字符串 (多个用,号分开, 异常情况有: 发烧,腹泻,呕吐,疥疮,咳嗽,
湿疹,咽喉痛)
13      "healthAskResult": "咳嗽,湿疹",
14      // 晨检合格 (体温检测合格 + 手部检测合格 + 健康询问合格)
15      // 晨检结果 是否健康 0 合格 1 不合格
16      "health": data.isAllHealth ? "0" : "1",
17
18      //下面 3 个是图片文件, 要接收保存。不是路径字符串!
19      // 人脸图片*1, 手部图片*2
20      "faceFile": faceFileList, //人脸图片,文件类型列表,list 里只有 1 条数据
21      "palmFile": palmFileList, // 手心图片,文件类型列表,list 里只有 1 条数据
22      "backOfHandFile": backOfHandFileList, // 手背图片,文件类型列表,list 里只有 1 条
数据
23  }
24
25
```

返回结果

变量名	字段名	是否必传	类型	示例值	描述
statusCode	请求状态码	是	String	200/300	200 为操作成功,其他为操作失败;
message	返回信息	是	string	"操作成功"	success 字段 true 成功提示的信息/success 字段 false 失败提示的信息;

返回结果格式如下：

```
1 // 提交成功
2 {
3   "statusCode":200,
4   "message":"操作成功",
5 }
6
7 // 提交失败
8 {
9   "statusCode":300,
10  "message": "",
11 }
12
```



3.轮询提交设备心跳信息

接口链接:

`${第三地服务域名或 ip 地址}/device/morningChecker/heartbeatInfo`

请求方法: POST

请求参数:

变量名	字段名	是否必传	类型	示例值	描述
equipmentCode /machineCode	设备 ID	是	String	6ac84eb75dcf44dc	设备的唯一 ID

提交数据范例如下：

```
1  {
2    "machineCode":"ac00146471d2c711",
3    "equipmentCode": "ac00146471d2c711"
4  }
5
```

返回结果

变量名	字段名	是否必传	类型	示例值	描述
statusCode	请求状态码	是	String	200/300	200 为操作成功,其他为操作失败;
message	返回信息	是	string	"操作成功"	success 字段 true 成功提示的信息/success 字段 false 失败提示的信息;

返回结果格式如下：

```
1 // 提交成功

2 {
3     "statusCode":200,
4     "message":"操作成功",
5 }
6
7 // 提交失败

8 {
9     "statusCode":300,
10    "message": "",
11 }
12
```

4. 获取健康咨询列表 (可选)

1. 返回的信息字段里面，必须有“健康”选项，即使没有，客户端收到数据后也会插入该选项

接口链接：

`${第三地服务域名或 ip 地址}/device/morning/getSymptomList`

请求方法: **GET**

请求参数:

变量名	字段名	是否必传	类型	示例值	描述
equipmentCode	设备 ID	是	String	6ac84eb75dcf44dc	设备的唯一 ID

返回结果格式如下：

```
1  // 提交成功
   {
2    "data":[
3      {"number":0,"symptomName":"健康"},//必须有健康选项
4      {"number":1,"symptomName":"发烧"},
5      {"number":2,"symptomName":"腹泻"}
6    ],
7    "success":true,
8    "resCode":200,
9    "message":"操作成功",
10   "resMsg":"操作成功",
11   "statusCode":200
12  }
13
14 // 提交失败
   {
15   "statusCode":300,
16   "message":"",
17  }
18
```

5. 上传本地注册的人脸图片（可选）

1. 本地添加人脸不上传服务器可以不对接
2. 更新 API 人脸注册需要 2 个步骤：下拉更新员工信息---> 点击“API 人脸图片注册”
3. 下拉刷新只更新员工信息，不进行 API 人脸注册
4. 刷新员工信息后，对比之前存在的注册图片有删减的话，客户端则会把之前注册的图片删除
5. 如果要本地注册上传服务器，点击添加人脸，在拍照注册成功后，会弹框提示“是否上传人脸图片”，确定的话就会上传图片，后台生成图片链接，更新到员工信息的 portraitPhoto 字段

接口链接:

`${第三地服务域名或 ip 地址}/device/face/employeeupload`

请求方法: **POST**

Content-Type	multipart/form-data
--------------	---------------------

请求参数:

变量名	字段名	是否必传	类型	示例值	描述
deviceId	设备 ID	是	String	6ac84eb75d cf44dc	设备的唯一 ID
userId	用户 ID	是	String	1001	
multipartFiles	图片文件	是	File[]		人脸图片文件列表 (仅支持 JPG 格式) list 里只有 1 条数据

返回结果格式如下 :

```
1 // 提交成功

2 {
3     "statusCode":200,
4     "message":"操作成功",
5 }
6
7 // 提交失败

8 {
9     "statusCode":300,
10    "message": "",
11 }
12
```