**Perfxlab人脸识别服务端接口使用说明**

# 概述

本系统利用REST技术实现前后端系统分离，相比较传统的服务器端渲染的方式，系统不仅具有更高的并行开发能力，并且页面都在客户端渲染，提高了渲染速度，减小了服务器的压力，一套API提供给多个客户端使用，展现了优越的跨平台能力。

本系统共包括四个主要模块：人员管理、部门管理、匹配记录管理以及终端管理。系统对每个模块都提供了增删查改的API接口，使得用户具备完全意义上的二次开发能力。

# 人员管理

**描述**

查询服务器添加的所有人员的详细信息，支持增删查改以及分页等操作。

**调用URL**

<http://IP:端口/api/v1/people/>

**查询调用方法**

GET

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 否 | limit | 每页显示多少人员信息 |
| 否 | offset | 从第几条记录开始查询 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| count | 共计拥有多少个人员信息 |
| next | 下一页链接地址 |
| pervioues | 上一页链接地址 |
| results | 所有人员的详细信息 |
| pk | 人员记录的主键 |
| uuid | 人员唯一标识 |
| position | 人员职位 |
| name | 人员姓名 |
| employee\_number | 员工号 |
| image | base64格式的图片 |
| urls | 查询、更新以及删除的操作链接地址 |
| is\_active | 在职或是离职状态 |
| employment\_date | 入职时间 |
| departure\_date | 离职时间 |
| department\_name | 所在部门名称 |
| devices | 所在终端主键 |

**请求成功返回示例**

{

"count": 36,

"next": "http://192.168.1.59:10011/api/v1/people/?limit=2&offset=4",

"previous": "http://192.168.1.59:10011/api/v1/people/?limit=2",

"results": [

{

"pk": 5,

"uuid": "f111527a-3c9f-11e9-a715-0242ac14000a",

"name": "测试人员1",

"position": "深度学习开发工程师",

"employee\_number": "P0004",

"image": "/9j/4AAQSkZJRg3jZ/mDKxO....................",

"urls": "/faces/detail/5/,/faces/update/5/,/faces/delete/5/",

"is\_active": true,

"employment\_date": "2017-04-25",

"departure\_date": null,

"department\_name": "软件部",

"devices": [

1

]

},

{

"pk": 1,

"uuid": "f1114758-3c9f-11e9-a715-0242ac14000a",

"name": "测试人员2",

"position": "CEO",

"employee\_number": "P0000",

"image": "/9j/4AAQSkZJRgABAQEASABIAAD/4Q...............",

"urls": "/faces/detail/1/,/faces/update/1/,/faces/delete/1/",

"is\_active": true,

"employment\_date": "2016-06-01",

"departure\_date": null,

"department\_name": "软件部",

"devices": [

1

]

}

]

}

**增加调用方法**

POST

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | name | 人员姓名 |
| 是 | position | 人员职位 |
| 是 | employee\_number | 员工号 |
| 是 | image | base64格式的图片 |
| 是 | is\_active | 在职或是离职 |
| 是 | employment\_date | 入职时间 |
| 否 | department\_name | 离职时间 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

**删除调用方法**

DELETE

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | pk | 人员主键 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

**修改调用方法**

PUT/PATCH

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | name | 人员姓名 |
| 是 | position | 人员职位 |
| 是 | employee\_number | 员工号 |
| 是 | image | base64格式的图片 |
| 是 | is\_active | 在职或是离职 |
| 是 | employment\_date | 入职时间 |
| 否 | department\_name | 离职时间 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

**描述**

将人员绑定要指定的终端设备上，支持绑定多个设备。

**绑定设备URL**

<http://IP:端口/faces/set_devices/>

**调用方法**

以POST方式发送form-data数据

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | person\_pk | 人员主键 |
| 是 | devices | 设备主键 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

# 3部门管理

**描述**

查询服务器添加的所有部门的详细信息，支持增删查改以及分页等操作。

**调用URL**

[http://IP:端口/api/v1/department/](http://IP:端口/api/v1/people/)

**查询调用方法**

GET

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 否 | limit | 每页显示多少部门信息 |
| 否 | offset | 从第几条记录开始查询 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| count | 共计拥有多少个部门信息 |
| next | 下一页链接地址 |
| pervioues | 上一页链接地址 |
| results | 所有部门的详细信息 |
| pk | 部门记录的主键 |
| name | 部门名称 |
| urls | 查询、更新以及删除的操作链接地址 |

**请求成功返回示例**

{

"count": 2,

"next": null,

"previous": null,

"results": [

{

"pk": 1,

"name": "硬件部",

"urls": "/departments/detail/1/,/departments/update/1/,/departments/delete/1/"

},

{

"pk": 2,

"name": "软件部",

"urls": "/departments/detail/2/,/departments/update/2/,/departments/delete/2/"

},

]

}

**增加调用方法**

POST

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | name | 部门名称 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

**删除调用方法**

DELETE

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | pk | 部门主键 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

**修改调用方法**

PUT/PATCH

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | name | 部门名称 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

# 4.匹配记录管理

**描述**

查询终端向服务端发送的匹配记录的详细信息，支持增删查改以及分页等操作。

**调用URL**

[http://IP:端口/api/v1/records/](http://IP:端口/api/v1/people/)

**查询调用方法**

GET

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 否 | limit | 每页显示多少匹配信息 |
| 否 | offset | 从第几条记录开始查询 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| count | 共计拥有多少个匹配信息 |
| next | 下一页链接地址 |
| pervioues | 上一页链接地址 |
| results | 所有匹配记录的详细信息 |
| pk | 匹配记录的主键 |
| target | 人员主键 |
| created | 检测时间 |
| record\_type | 记录类型 |
| target\_\_name | 人员姓名 |
| target\_\_position | 人员职位 |
| target\_\_employee\_number | 员工号 |
| target\_\_department | 人员部门 |

**请求成功返回示例**

{

"count": 1,

"next": null,

"previous": null,

"results": [

{

"pk": 1,

"target": 23,

"created": "2019-02-02 05:02:00",

"record\_type": "IN",

"target\_\_name": "测试",

"target\_\_position": "算法工程师",

"target\_\_employee\_number": "F0002",

"target\_\_department": "软件部"

}

]

}

**增加调用方法**

POST

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | target | 人员主键 |
| 是 | created | 检测时间 |
| 是 | record\_type | 记录类型 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

**删除调用方法**

DELETE

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | pk | 记录主键 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}

# 终端管理

**描述**

查询所有已经向服务端注册过的终端信息。终端启动时会向固定IP自动注册，不需要后台系统进行增删改等操作。

**查询URL**

<http://IP:端口/api/v1/devices/>

**调用方法**

GET

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 否 | limit | 每页显示多少匹配信息 |
| 否 | offset | 从第几条记录开始查询 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| count | 共计拥有多少个设备信息 |
| next | 下一页链接地址 |
| pervioues | 上一页链接地址 |
| results | 所有设备的详细信息 |
| pk | 设备的主键 |
| uuid | 设备的唯一标识 |
| name | 设备名称 |
| description | 设备描述 |
| device\_type | 终端类型 |
| ip | 终端ip |
| members | 终端绑定人员的主键 |

**请求成功返回示例**

{

"count": 1,

"next": null,

"previous": null,

"results": [

{

"pk": 1,

"uuid": "e2ce74c6-1276-11e9-b954-309c23d8cbc7",

"name": "测试设备",

"descriptions": null,

"device\_type": "IN",

"position": "后门",

"ip": "127.0.0.1",

"members": []

}

]

}

**终端绑定人员URL**

<http://IP:端口/sync/devices/set_people/>

**调用方法**

以POST方式发送form-data数据

**请求参数**

|  |  |  |
| --- | --- | --- |
| **是否必选** | **参数名** | **参数说明** |
| 是 | device\_pk | 设备主键 |
| 是 | people | 人员主键 |

**返回值说明**

|  |  |
| --- | --- |
| **字段** | **说明** |
| error | 执行结果状态 |
| detail | 详细执行结果 |

**请求成功返回示例**

{

“error”:false

}