

一些简单的约定

返回码：

- 200: 正常操作
- 201: 创建新资源
- 400: 不好的请求，比如需要的字段没有，时间错误，管理别人的 holiays，相应错误会在返回的message
- 401: 没有认证，在登陆认证前就进行操作，需要先登录，然后带 token
- 403: 禁止，没有对应权限

操作：

- PUT: 更新一个已经有的资源
- POST: 创建一个新的资源
- GET: 获得对应的资源

GET请求查询时，都做了分页：

- page: 第几页
- per_page: 每一页有多少
- want_all: 要全部的页

对应接口

普通操作

holiday_apply_state: 0:正在等待审批, -1:取消中, 1:在一级审批, 2: 在2级审批, 3: 在3级审批

holiday_apply_ok: 0: 在审批中, -1: 被驳回, 1:审批成功

holiday_app_over: 默认为0。 1: 假最终结束

holiday_app_end: 默认为0。 1:申请销假

holiday_type:

1. 'Leave on business'
2. 'Annual leave'
3. 'Sick leave'
4. 'Funeral leave'
5. 'compassionate leave'
6. 'Maternity leave'
7. 'Marriage leave'
8. 'Maternity leave'

用户登录

GET 服务器ip: 服务器端口/api/v1.0/token

auth中: username为账号, password: 为密码

返回

expiration: 为token有效的时间, 单位为s

token: 为token, 在之后的请求中带有auth, 将username中的值变成token

```
{
  "expiration": 3600,
  "token":
  "eyJleHAiOjE0ODA3NDk2MjcslmFsZyI6IkhTMjU2IiwiaWF0IjoxNDgwNzQ2MDI3fQ.eyJpZCI6IjMwMTMyMTgweHgifQ.berBE7ELji1KnJr0IZ0QiAdsPJufh8Eh5eT0Gb79fY"
}
```

用户看自己的请的假

GET 服务器ip: 服务器端口/api/v1.0/worker/holidays

需要在auth中带有token,

如:eyJleHAiOjE0ODA3NDk2MjcslmFsZyI6IkhTMjU2IiwiaWF0IjoxNDgwNzQ2MDI3fQ (这个一直在变)

返回:

holidays为你请假的数组, 然后里面为你请的假

holiday_worker_id: 为请假者的id, 这就是你

holiday_type: 请假的类型

holiday_time_end: 请假的结束时间

holiday_time_begin: 请假的开始时间

holiday_reason: 请假的理由

holiday_id: 这次假标识的ID

holiday_end_time: 这次假, 最终销假的时间

holiday_apply_time: 这次假申请的时间

holiday_apply_state: 这次假处在那个级别审批中的状态

holiday_apply_ok: 这次假的申请状态

holiday_app_over: 这次假是否全部结束了

holiday_app_end: 是否申请销假

```
{
  "holidays": [
    {
      "holiday_app_end": false,
      "holiday_app_over": false,
      "holiday_apply_ok": 0,
      "holiday_apply_state": 0,
      "holiday_apply_time": null,
      "holiday_end_time": null,
      "holiday_id": 1,
      "holiday_reason": "just haha",
      "holiday_time_begin": "2016-10-01",
      "holiday_time_end": "2016-10-02",
      "holiday_type": 4,
      "holiday_worker_id": "30132180xx"
    },
    {
      "holiday_app_end": false,
      "holiday_app_over": false,
      "holiday_apply_ok": 0,
      "holiday_apply_state": 0,
      "holiday_apply_time": null,
      "holiday_end_time": null,
      "holiday_id": 2,
      "holiday_reason": "just haha",
      "holiday_time_begin": "2016-10-01",
      "holiday_time_end": "2016-10-10",
      "holiday_type": 2,
      "holiday_worker_id": "30132180xx"
    },
  ],
}
```

```
{
  "holiday_app_end": false,
  "holiday_app_over": false,
  "holiday_apply_ok": 0,
  "holiday_apply_state": 0,
  "holiday_apply_time": "Thu, 01 Dec 2016 00:00:00
GMT",
  "holiday_end_time": null,
  "holiday_id": 5,
  "holiday_reason": "just test",
  "holiday_time_begin": "2016-10-01",
  "holiday_time_end": "2016-10-02",
  "holiday_type": 4,
  "holiday_worker_id": "30132180xx"
}
]
```

用户申请新的假

POST 服务器ip: 服务器端口/api/v1.0/worker/holidays

请求示范:

对应字段意义和开始约定一样

需要注意, 一定要有reason, begin要早于end

```
{
  "holiday_type": "4",
  "holiday_time_begin": "2016-10-1",
  "holiday_time_end": "2016-10-22",
  "holiday_reason": "just test"
}
```

返回:

申请成功，创建的holidays的id

```
{  
  "holiday_id": 7  
}
```

如果开始比结束晚的话：

```
{  
  "error": "bad request",  
  "message": "begin is latter than end"  
}
```

没有理由或者理由为空：

```
{  
  "error": "bad request",  
  "message": "you need reason to apply"  
}
```

请假类型不存在：

```
{  
  "error": "bad request",  
  "message": "the holiday type not exit"  
}
```

如果是为请的年假，超过时间的话：

目前为一申请年假马上扣掉年假时间，只有这个假被驳回，然后你放弃申请才能重新加上去，或者还没审批时候，你取消申请

```
{  
  "error": "bad request",  
  "message": "your annual leave nums isn't enough"
```

```
}
```

用户修改请假申请

注意修改只能在还没被审批时和申请被驳回时才能修改。

申请结束必须在这次假已经成功申请之后才能申请。

PUT 服务器ip: 服务器端口/api/v1.0/worker/holidays / holiday的id

、

请求修改例子如下:

注意over是修改直接放弃这次请假

```
{
  "holiday_over": "1"
}
```

下面是修改一个假的信息

```
{
  "holiday_reason": "just haha",
  "holiday_begin_time": "2016-10-1",
  "holiday_end_time": "2016-10-20",
  "holiday_type": "3",
  "holiday_over": "1"
}
```

下面是当你审批ok之后, 放完假, 回来工作了, 要申请销假

```
{
  "holiday_end": "1"
}
```

返回:

成功时

```
{  
  "message": "your modify your holiday apply"  
}
```

管理一个不存在的holiday

```
{  
  "error": "bad request",  
  "message": "the holiday isn't exist"  
}
```

管理不是你的holiday

```
{  
  "error": "bad request",  
  "message": "your can't modify the holiday which is  
not belong you"  
}
```

修改一个已经审批成功的holiday:

```
{  
  "error": "bad request",  
  "message": "you can't modify the apply ok holiday"  
}
```

修改一个正在审批的holiday:

```
{  
  "error": "bad request",  
  "message": "you can't modify the holiday in check"
```



```
}
```

其他类似申请创建一个新holiday一样

用户申请结束自己的假期

申请销假：

PUT 服务器ip: 服务器端口/api/v1.0/worker/holidays / holiday的id

```
{  
  
  "holiday_end" : "1"  
}
```

成功

```
{  
  "message": "your apply to end the holiday"  
}
```

假没有审核成功

```
{  
  "error": "bad request",  
  "message": "your can apply to end the holiday maybe  
in check or apply faily"  
}
```

用户查看自己加班申请的信息

GET 服务器ip: 服务器端口/api/v1.0/worker/workadds

,

返回：

同holidays, 如果没有为空数组

```
{
  "workadds": [
    {
      "workadd_add_state": 0,
      "workadd_apply_time": "Sat, 03 Dec 2016 00:00:00
GMT",
      "workadd_end_time": "Sun, 02 Oct 2016 00:00:00 GMT",
      "workadd_id": 2,
      "workadd_reason": "just like work",
      "workadd_start_time": "Sat, 01 Oct 2016 00:00:00
GMT"
    },
    {
      "workadd_add_state": 0,
      "workadd_apply_time": "Sat, 03 Dec 2016 00:00:00
GMT",
      "workadd_end_time": "Sun, 02 Oct 2016 00:00:00 GMT",
      "workadd_id": 3,
      "workadd_reason": "just like work",
      "workadd_start_time": "Sat, 01 Oct 2016 00:00:00
GMT"
    },
    {
      "workadd_add_state": 0,
      "workadd_apply_time": "Sat, 03 Dec 2016 00:00:00
GMT",
      "workadd_end_time": "Sun, 02 Oct 2016 00:00:00 GMT",
      "workadd_id": 4,
      "workadd_reason": "just like work",
      "workadd_start_time": "Sat, 01 Oct 2016 00:00:00
GMT"
    },
  ],
}
```

```
{
  "workadd_add_state": 0,
  "workadd_apply_time": "Sat, 03 Dec 2016 00:00:00
GMT",
  "workadd_end_time": "Sun, 02 Oct 2016 00:00:00 GMT",
  "workadd_id": 5,
  "workadd_reason": "just like work",
  "workadd_start_time": "Sat, 01 Oct 2016 00:00:00
GMT"
}
]
```

创建新的加班

POST`服务器ip: 服务器端口/api/v1.0/worker/workadds

```
{
  "workadd_time_begin": "2016,10,1",
  "workadd_time_end": "2016,10,2",
  "workadd_reason": "just like work"
}
```

成功时:

```
{
  "workadd_id": 6
}
```

时间开始比结束晚:

```
{
  "error": "bad request",
  "message": "workadd start later than end"
```

```
}
```

没有理由或者理由为空字符串

```
{  
  "error": "bad request",  
  "message": "workadd must have reason"  
}
```

修改加班信息

PUT 服务器ip: 服务器端口/api/v1.0/worker/workadds / workadd的id

```
{  
  
  "workadd_reason": "just tkjf",  
  "workadd_start": "2016-10-1",  
  "workadd_end": "2016-10-2"  
}
```

成功

```
{  
  "message": "you modify your workadd"  
}
```

时间错误:

```
{  
  "error": "bad request",  
  "message": "the start latter than end"  
}
```

管理操作

以下操作需要对应的权限

查看自己目前需要管理的申请的holidays

GET 服务器ip: 服务器端口/api/v1.0/examine/holidays

、

下面字段含义和上面的一样

返回:

```
{
  "holidays": [
    {
      "holiday_app_end": false,
      "holiday_app_over": false,
      "holiday_apply_ok": 0,
      "holiday_apply_state": 2,
      "holiday_apply_time": "Sat, 03 Dec 2016 00:00:00 GMT",
      "holiday_end_time": null,
      "holiday_id": 6,
      "holiday_reason": "just test",
      "holiday_time_begin": "2016-10-01",
      "holiday_time_end": "2016-10-22",
      "holiday_type": 4,
      "holiday_worker_id": "30132180xx"
    },
    {
      "holiday_app_end": false,
      "holiday_app_over": false,
      "holiday_apply_ok": 0,
      "holiday_apply_state": 2,
      "holiday_apply_time": "Sat, 03 Dec 2016 00:00:00 GMT",

```

```

"holiday_end_time": null,
"holiday_id": 7,
"holiday_reason": "just test",
"holiday_time_begin": "2016-10-01",
"holiday_time_end": "2016-10-22",
"holiday_type": 4,
"holiday_worker_id": "30132180xx"
},
{
"holiday_app_end": true,
"holiday_app_over": true,
"holiday_apply_ok": 1,
"holiday_apply_state": 1,
"holiday_apply_time": null,
"holiday_end_time": null,
"holiday_id": 4,
"holiday_reason": "adsasdad",
"holiday_time_begin": "2016-11-09",
"holiday_time_end": "2016-11-12",
"holiday_type": 1,
"holiday_worker_id": "3013218077"
}
]
}

```

审批对应的申请holiday

PUT`服务器ip: 服务器端口/api/v1.0/examine/holidays/holiday的id/check

```

{
"holiday_ok": "-1"
}

```

返回:

成功:

```
{  
  "message": "examine ok"  
}
```

已经审核了:

```
{  
  "error": "bad request",  
  "message": "holiday not in check"  
}
```

管理一个已经取消

```
{  
  "error": "bad request",  
  "message": "cancel the holiday"  
}
```

管理不存在的holiday

```
{  
  "error": "bad request",  
  "message": "don't exit the holiday"  
}
```

管理一个不属于你现在管理的或者你没有权限管理的

```
{  
  "error": "bad request",  
  "message": "your can't examine the holidays"
```

```
}
```

审批对应的销假

PUT`服务器ip: 服务器端口/api/v1.0/examine/holidays/holiday的id/over

`

```
{  
  "holiday_over": "1"  
}
```

成功:

```
{  
  "message": "ok"  
}
```

不存在的holiday:

```
{  
  "error": "bad request",  
  "message": "don't exit the holiday"  
}
```

已经over的holiday:

```
{  
  "error": "bad request",  
  "message": "this holiday is over"  
}
```

还有申请销假的holiday:

```
{
```



```
"error": "bad request",
"message": "holiday isn't over and wait to check
over"
}
```

没有权限去over的

```
{
"error": "bad request",
"message": "your can't examine the holidays"
}
```

查看自己需要审批的加班

GET 服务器ip: 服务器端口/api/v1.0/examine/workadds

、

返回

```
{
"workadds": [
{
"workadd_add_state": 0,
"workadd_apply_time": "Wed, 30 Nov 2016 22:53:47
GMT",
"workadd_end_time": "Sun, 02 Oct 2016 00:00:00 GMT",
"workadd_id": 1,
"workadd_reason": "just haha",
"workadd_start_time": "Sat, 01 Oct 2016 00:00:00
GMT"
}
]
}
```

管理下属加班

PUT 服务器ip: 服务器端口/api/v1.0/examine/workadds / 加班的id

```
{  
  "workadd_ok": "-1"  
}
```

返回

成功:

```
{  
  "message": "examine ok"  
}
```

已经审批过的

```
{  
  "error": "bad request",  
  "message": "the workadd is examine over"  
}
```

用户查看修改自己的信息

是在api/v1.0下面的

```
@api.route('/user/', methods=['PUT'])  
    json_worker = request.json  
    name = json_worker.get('worker_name')  
    email = json_worker.get('worker_email')  
    address = json_worker.get('worker_address')  
    password = json_worker.get('worker_password')  
    year_holidays_residue =
```

```

json_worker.get('worker_year_holidays_residue')
    year_holidays_used =
json_worker.get('worker_year_holidays_use')
    workAdd_time = json_worker.get('workAdd_time')

@api.route('/user/', methods=['GET'])
def get_worker_info():
    worker = g.current_user

    if worker is None:
        return bad_request('no exists the worker')

    return jsonify(worker.to_json())

```

admin

注意这是单独的另外一个url 比如127.0.0.1:5000/admin/xxxx

- @adminapi.route('/users/')
 - @adminapi.route('/users/', methods=['POST'](#)): 字段有:
worker_id,workername,worker_email,worker_address,password
 - @adminapi.route('/users/<string:id>/degree',
methods = [" P O S T "](#)), 字段有:
degree_department_id,degree_degree_id
 - @adminapi.route('/users/<string:id>', methods=['PUT'](#)), 字段
有: name =
worker_name,worker_email,worker_address,worker_passw

ord,worker_year_holidays_residue,worker_year_holidays_u
se,workAdd_time

- @adminapi.route('/users/<string:uid>/department/
<int:did>', methods=['PUT'](#))字段有:worker_degree_degree