

接口文档

一.扫码登陆

扫码采用内嵌到页面中的方式

appid: dingncwq2vipv3gcqakc

redirect_uri: http://{地址单独给}/api/zfsh/dingding/scanCodeLogin

测试案例

https://oapi.dingtalk.com/connect/qrconnect?

appid=dingncwq2vipv3gcqakc&response_type=code&scope=snsapi_login&state=STATE&redirect_uri=http://{地址单独给}/api/zfsh/dingding/scanCodeLogin

扫码成功返回消息体

```
{
  "code": "1",           1成功    为0失败
  "msg": "",
  "data": {
    "loginCode": "6131925a3e3f0762690f2a24",  登录凭证
    "staffInfo": {                             登录者信息
      "userName": "登录名",                    登录者名称
      "avatar": "",                             登录者头像(现在可能为空, 后面我会
默认给个头像地址)
      "userId": "1931403421838514"            登录者账号id
    }
  }
}
```

所有的code码 1表示请求成功 0表示失败

特殊的一些code码 比如-1 表示登录过期 需要返回登录界面

后期可能会有其他的特殊的code码, 到时候用到再加

二.部门接口

注意点: 目前除登录扫码不需要在请求头里传信息外, 后台系统所有的接口在请求头里添加 _wt:登录凭证
目前不加请求头我不会拦截, 后期如果没有 请求错误码为0 msg会返回非法的请求

get /api/coral/dept/listHierarchyDept

```
{
  "code": 1,
  "msg": null,
  "data": [
    {
      "deptId": "1",
      "deptName": "中福测试",
      "hasChild": true,  是否有子部门
      "child": [         子部门信息
        {

```

```

        "deptId": "505934047",
        "deptName": "产品部1",
        "hasChild": false,
        "child": null
      }
    ]
  }
}

```

三.员工信息分页

```

POST /api/coral/staff/pageInfo
{
  "phone": "18652493703",
  "userName": "名称",
  "staffStatus": "1",      1 在职  2离职
  "deptId": "2"           部门id   （左边点击选中某个部门，直接请求刷新员工分页接口）
}

```

```

{
  "code": 1,
  "msg": null,
  "data": {
    "total": 1,              总记录数
    "currentPage": 1,
    "totalPages": 1,        总页数
    "pageSize": 10,         分页数量
    "rows": [
      {
        "id": 1,
        "userId": "1931403421838514",
        "userName": "名称",
        "avatar": "",        头像
        "mobile": "18652493703",
        "title": "CEO",       职位
        "adminLogin": 1,      1 有权登录后台  2无权
        "staffStatus": 1,    1在职  2离职
        "gmtCreate": "2021-09-03T00:51:07"  日期我后面处理一下，直接展示就行
      }
    ]
  }
}

```

四.员工同步接口

```

get /api/coral/staff/sync

```

```

{
  "code": 1,
  "msg": null,
  "data": true
}

```

五.类目新增与修改

```

post /api/coral/category/modifyCategory
新增
{
  "categoryName": "梭子蟹",          类目名
  "parentCategoryId": "201201201"  父类目 一级类目可不传或-1
}
修改
{
  "categoryId": "202",              类目id
  "categoryName": "鞋子"           类目名
}

```

```

{
  "code": 1,
  "msg": null,
  "data": true
}

```

六.类目列表(所有目录按照层级返回)

```

get /api/coral/category/loopCategory

```

```

{
  "code": 1,
  "msg": null,
  "data": [
    {
      "categoryId": 201,
      "categoryName": "生鲜蔬菜",
      "hasChild": true,          是否有子目录
      "child": [                子目录
        {
          "categoryId": 201201,
          "categoryName": "海鲜",
          "hasChild": true,
          "child": [
            {
              "categoryId": 201201201,
              "categoryName": "螃蟹",
              "hasChild": true,
              "child": [
                {
                  "categoryId": 201201201201,
                  "categoryName": "梭子蟹",
                  "hasChild": false,
                  "child": null
                }
              ]
            }
          ]
        }
      ]
    }
  ],
  {
    "categoryId": 202,

```

```
        "categoryName": "鞋子",
        "hasChild": false,
        "child": null
    }
]
}
```

七.查询父类目下的子类目

```
post /api/coral/category/childCategory
{
  "parentCategoryId":"201201201"
}
```

```
{
  "code": 1,
  "msg": null,
  "data": [
    {
      "categoryId": 201201201201,
      "categoryName": "梭子蟹"
    }
  ]
}
```

八.删除类目

```
post /api/coral/category/delCategory
{
  "categoryId":""  类目id
}
```

```
{
  "code": 1,
  "msg": null,
  "data": true
}
```