模拟接口已在easymock上，里面有具体的接口地址以及命名与返回情况。

在 <https://www.easy-mock.com> 注册后，把用户名发给我，我邀请你协作，就可以一起编辑接口了。

由于采用 Vue的axios不支持jsonp，接口目前需要在服务端配置CORS

------------------------------------------------------------------------------------------

登录验证并获取token:

POST /login/login.do

传入参数:

username|用户名|string|必须|

password|密码|string|必须|

传出参数(成功)：

{

code:200,

token: ‘admin’ //token用于之后查询用户数据

}

传出参数(密码不正确)：

{

code:401

}

传出参数(服务器错误):

{

code:500

}

------------------------------------------------------------------------------------------

获取个人信息:

GET /user/info.do

传入参数:

token| token串 |string|必须|

传出参数(成功获取)：

{

code: 200,

content: {

role: [“admin”], //role为数组，里面只有admin和employee两种角色

token: "admin",

avatar: "https://wpimg.wallstcn.com/f778738c-e4f8-4870-b634-56703b4acafe.gif", //头像

name: "Super Admin" //用户名

}

}

------------------------------------------------------------------------------------------

退出登录

GET /login/logout.do

传入参数:

传出参数(退出成功)：

{

code:200

}

------------------------------------------------------------------------------------------

获取菜品列表信息: （带筛选和查询功能）

GET /food/foodList.do

传入参数:

page| 当前点击页码 |Number|必须|

limit| 一页取10个|Number|必须|

title| 要查询的菜品名字|String|不必须|

type|要筛选的分类名称 |String|不必须|

传出参数(成功获取)：

{

"code": 200,

“total”:40, // 菜品总数。

"items": [

{

"foodId": 2645, //菜品ID

"title": "手撕鸡", //菜品名称

"img": "http://dummyimage.com/430x430", //菜品预览图

"price": 9.12, //单价原价

"offprice": 87.35, //折后价

"type": "XC", // 菜品类别

"status": "offshelf", //上下架状态

"ordernum": 2992 //点菜次数

},

{

"foodId": 1009,

"title": "干锅包菜",

"img": "http://dummyimage.com/430x430",

"price": 83.86,

"offprice": 82.67,

"type": "CX",

"status": "onshelf",

"ordernum": 1825

}]

}

------------------------------------------------------------------------------------------

更新单个菜品信息:

POST /update/updateFood.do

传入参数:

foodId|菜品id |Number |必须|

title|菜名|string|必须|

img | 预览图| string | 必须|

type |分类 | string | 必须 | // "YC"、"XC"、"HB"、"CX" 粤菜、西餐、杭帮、川湘

price |价格 | Number | 必须 |

offprice |折后价 | Number | 必须 |

status |状态 | string| 必须 | // ”offshelf” 、 “onshelf”

传出参数(成功获取)：

{

code:200

}

------------------------------------------------------------------------------------------

增加菜品 ：

POST /add/addFood.do

传入参数:

foodId|菜品id |Number |不必须 | //传过来的时候是id应该是空的，id由后端返回

title|菜名|string|必须|

img | 预览图| string | 必须|

type |分类 | string | 必须 | // "YC"、"XC"、"HB"、"CX" 粤菜、西餐、杭帮、川湘

price |价格 | Number | 必须 |

offprice |折后价 | Number | 必须 |

status |状态 | string| 必须 | // ”offshelf” 、 “onshelf”

传出参数(成功获取)：

{

code: 200，

foodId: 1330

}

------------------------------------------------------------------------------------------

删除菜品:

POST /delete/deleteFood.do

传入参数:

foodId|菜品id |Number |必须 |

传出参数(删除成功)：

{

code: 200

}

------------------------------------------------------------------------------------------

切换菜品上下架状态:

POST /update/foodStatus.do

传入参数:

foodId|菜品id |Number |必须|

status|状态|string|必须| // offshelf 、onshlef

传出参数(切换成功)：

{

code: 200

}

------------------------------------------------------------------------------------------

点餐页获取菜单: (与之前菜品列表不同，返回的是上架的菜品信息) 不做筛选和查询

GET /food/getfoodList.do

传入参数:

传出参数(切换成功)：

{

"code": 200,

"items": [

{

"foodId": 2892,

"title": "手撕鸡",

"price": 56.96,

"offprice": 65.32,

"type": "YC"

},

{

"foodId": 1805,

"title": "干锅包菜",

"price": 92.35,

"offprice": 69.84,

"type": "CX"

}]

}

------------------------------------------------------------------------------------------

获取空位的桌号：(即已结账的桌号)

GET /desk/getDesk.do :

传入参数

传出参数(获取成功)：

{

code:200,

"content": [1,2,4,7,9,16,24,30]

}

------------------------------------------------------------------------------------------

下单接口：

POST /food/droporder.do:

传入参数:

foods |菜品数组 |Array|必须|

deskId|桌号 |Number |必须

传出参数(下单成功)：

{

code:200

}

------------------------------------------------------------------------------------------

获取订单列表：

GET /order/orderList.do

传入参数：

传出参数(获取成功)：

{

"code": 200,

"items": [

{

"orderId": 3189, //订单id

"deskId": 18, // 桌号

"timestamp": "2017-04-09 01:52:28", //下单时间

"employee": "毛刚", //下单负责的服务员

"money": 30.37, //应付总额

"status": 2 //订单状态 1：未付款 2：已付款 3：已取消

},

{

"orderId": 4537,

"deskId": 15,

"timestamp": "2002-03-02 11:01:13",

"employee": "武杰",

"money": 28.41,

"status": 2

},

{

"orderId": 4198,

"deskId": 9,

"timestamp": "2003-07-25 08:44:47",

"employee": "张秀英",

"money": 25.23,

"status": 1

}]

}

------------------------------------------------------------------------------------------

订单详情：

GET /order/orderDetail.do

传入参数:

传出参数(下单成功)：

{

"code": 200,

"goodList": [

{

"id": 100001, //菜品id

"title": "菜品1", //菜品名称

"url": "http://yanxuan.nosdn.127.net/b2d8936707338e5c734f000e1597e930.jpg?imageView&thumbnail=430x430", //图片

"unitPrice": 10, //单价 这里如果offprice折后价低于单价，就取折后价

"quantity": 10, //数量10

"total": 100 //总计金额

},

{

"id": 100002,

"title": "商品2",

"url": "http://yanxuan.nosdn.127.net/b2d8936707338e5c734f000e1597e930.jpg?imageView&thumbnail=430x430",

"status": 1,

"unitPrice": 100,

"quantity": 20,

"total": 2000

}

],

"status": 1,

"deskId": 20

}

------------------------------------------------------------------------------------------

获取员工列表：

GET /employee/employeeList.do

传入参数:

传出参数(获取成功)：

{

"code": 200,

"total": 40,

"items": [

{

"employeeId": 1833, //员工id

"name": "汤娟", //员工名称

"age": 29, // 年龄

"mobile": "14111816319", //手机号

"salary": 1291.39 // 薪资

},

{

"employeeId": 1645,

"name": "金丽",

"age": 22,

"mobile": "18359318152",

"salary": 1668.51

}]

}

------------------------------------------------------------------------------------------

增加员工接口：

POST /add/addEmployee.do

传入参数:

employeeId| 员工|Number |不必须| 传过去应该是空的，id由后端返回生成

name|员工姓名|String |必须

age|员工年龄| String |必须

mobile|手机|String |必须

salary|薪资|String |必须

传出参数(增加成功)：

{

"code": 200,

"id": 1411

}

------------------------------------------------------------------------------------------

更新员工接口：

POST /update/updateEmployee.do

传入参数:

employeeId| 员工|Number |必须|

name|员工姓名|String |必须

age|员工年龄| String |必须

mobile|手机|String |必须

salary|薪资|String |必须

传出参数(更新成功)：

{

"code": 200

}

------------------------------------------------------------------------------------------

删除员工接口：

POST /delete/deleteEmployee.do

传入参数:

employeeId| 员工|Number |必须|

传出参数(删除成功)：

{

"code": 200

}

------------------------------------------------------------------------------------------

上传图片接口：

POST /uploadImg.do

传入参数:

组件封装不需要填写参数

传出参数(上传成功)：

{

"cdnURL": [

{

"filename": "IMG20171025\_153606.png",

"remote": "http://mimg.127.net/hz/uploader/20171107/15100464942165330.png"

}

],

"code": 200

}