

# RESTful Api 接口设计以及 egg-jwt 实现 接口权限验证

主讲教师：（大地）

合作网站：[www.itying.com](http://www.itying.com) （IT 营）

我的专栏：<https://www.itying.com/category-79-b0.html>

1、首页菜品 Api 接口.....	1
2、菜品详情 Api 接口.....	2
3、购物车列表 Api 接口.....	3
4、加入购物车 Api 接口.....	4
5、增加购物车数量 Api 接口.....	5
6、减少购物车数量 Api 接口.....	5
7、获取桌子对应的购物车数量.....	6
8、获取口味信息.....	6
9、增加/修改用户用餐信息.....	7
10、 获取用户用餐信息.....	7
11、 提交订单.....	8
12、根据桌号获取已下单菜品 Api 接口.....	9
13、egg-jwt 实现接口权限验证.....	10

## 1、首页菜品 Api 接口

请求方式：get

地址：域名[/api/v1/productList](#)

返回参数说明：

参数	说明	备注
id	分类 id	

title	分类标题	
description	上级分类 id	
Products	菜品列表（见菜品参数说明）	

#### 菜品参数说明：

参数	说明	备注
id	菜品 id	
cid	分类 id	
title	菜品标题	
imgUrl	图片地址	
price	价格	
sort	排序	

## 2、菜品详情 Api 接口

请求方式：get

地址：域名 [/api/v1/productContent](#)

传递参数：

参数	说明	备注
id	菜品 id	必传

返回参数说明:

参数	说明	备注
id	菜品 id	
cid	分类 id	
title	菜品标题	
price	价格	
imgUrl	图片地址	
content	内容	
saleCount	销售数量	
isBest	精选	
isHot	热销	
sort	排序	

### 3、购物车列表 Api 接口

请求方式: get

地址: 域名 [/api/v1/cartList](#)

传递参数:

参数	说明	备注
tableId	桌号 (一个桌子有多位用户, 他们的桌号一样)	必传

返回参数:

参数	说明	备注
success	true/false	成功失败

result	集合	集合
--------	----	----

**result 参数说明:**

参数	说明	备注
tableId	桌号 （一个桌子有多位用户，他们的桌号一样）	
productTitle	菜品名称	
productId	菜品 id	
imgUrl	菜品图片	
price	菜品价格	
productNum	菜品数量	

## 4、加入购物车 Api 接口

请求方式: post

地址: 域名 [/api/v1/addCart](#)

传递参数:

参数	说明	备注
tableId	桌号 （一个桌子有多位用户，他们的桌号一样）	必传 （001 002）
productTitle	菜品名称	必传
productId	菜品 id	必传
imgUrl	菜品图片	必传
price	菜品价格	必传
productNum	菜品数量	必传

返回参数:

参数	说明	备注
success	true/false	成功失败
msg	说明信息	说明信息

## 5、增加购物车数量 Api 接口

请求方式: put

地址: 域名/[api/v1/incCart](#)

传递参数:

参数	说明	备注
tableId	桌号 (一个桌子有多位用户, 他们的桌号一样)	必传 (001 002)
productId	菜品 Id	必传

返回参数说明:

参数	说明	备注
success	true/false	成功/失败
msg	提示信息	提示信息

## 6、减少购物车数量 Api 接口

请求方式: put

地址: 域名/[api/v1/decCart](#)

传递参数:

参数	说明	备注
tableId	桌号 (一个桌子有多位用户, 他们	必传 (001 002)

	的桌号一样)	
productId	菜品 Id	必传

返回参数说明:

参数	说明	备注
success	true/false	成功/失败
msg	提示信息	提示信息

## 7、获取桌子对应的购物车数量

请求方式: get

地址: 域名/[api/v1/cartCount](#)

传递参数:

参数	说明	备注
tableId	桌号 (一个桌子有多位用户, 他们的桌号一样)	必传 (001 002)

返回参数说明:

参数	说明	备注
success	true/false	成功/失败
result	数量	

## 8、获取口味信息

请求方式: get

地址: 域名/[api/v1/flavorList](#)

传递参数: 无

返回参数说明:

参数	说明	备注
success	true/false	成功/失败
result	口味列表	

## 9、增加/修改用户用餐信息

请求方式: get

地址: 域名/api/v1/addPeopleInfo

传递参数:

参数	说明	备注
tableId	桌号 (一个桌子有多位用户, 他们的桌号一样)	必传 (001 002)
pNum	用餐人数	
pMark	口味备注	

返回参数说明:

参数	说明	备注
success	true/false	成功/失败
msg	提示信息	提示信息

## 10、获取用户用餐信息

请求方式: get

地址: 域名/api/v1/getPeopleInfo

传递参数:

参数	说明	备注
tableId	桌号 (一个桌子有多位用户, 他们的桌号一样)	必传 (001 002)

返回参数说明:

参数	说明	备注
success	true/false	成功/失败
result	用户用餐信息	用户用餐信息

## 11、提交订单

请求方式: post

地址: 域名/api/v1/addOrder

传递参数说明:

参数	说明	备注
tableId	桌号 (一个桌子有多位用户, 他们的桌号一样)	必传 (001 002)
pNum	用餐人数	必传
pMark	备注信息	必传
orderItems	菜品信息 (数组, 见菜品参数说明)	序列化的数组
totalPrice	总价格	总价格
totalNum	总数量	总数量

orderItem 菜品参数说明:

参数	说明	备注
----	----	----



productId	菜品 Id	
productTitle	菜品标题	
productPrice	菜品价格	
productNum	菜品数量	
productImg	菜品图片	

返回参数说明:

参数	说明	备注
success	true/false	订单 id
msg	说明	

## 12、根据桌号获取已下单菜品 Api 接口

请求方式: get

地址: 域名/api/v1/getOrder

传递参数:

参数	说明	备注
tableId	桌号	必传

返回参数说明:

参数	说明	备注
tableId	桌号 id	
pNum	用餐人数	

pMark	备注信息	
orderItems	数组，见菜品参数说明	
totalPrice	总价格	总价格
totalNum	总数量	总数量
orderStatus	订单状态	0 表示已下单并且未支付 1 表示已支付 已完结 2 表示取消
payStatus	支付状态	0 表示未支付 1 表示已经支付

#### orderItems 菜品参数说明:

参数	说明	备注
productId	菜品 Id	
productTitle	菜品标题	
productPrice	菜品价格	
productNum	菜品数量	
productImg	菜品图片	
status	菜品 Id	1 表示已经下厨 2 表示退菜

## 13、egg-jwt 实现接口权限验证

### 1、安装 Jwt

```
npm i egg-jwt --save
```

## 2、配置插件 {app\_root}/config/plugin.js

```
exports.jwt = {  
  enable: true,  
  package: "egg-jwt"  
};
```

## 3、配置签名 {app\_root}/config/config.default.js

```
exports.jwt = {  
  secret: "123456"  
};
```

## 4、生成 token

```
async login(){  
  const token = this.app.jwt.sign({ foo: 'bar' }, this.app.config.jwt.secret,{  
    expiresIn: 60*60*2  
  });  
  this.ctx.body=token;  
}
```

## 5、配置路由

在需要使用 Jwt 验证的路由上面配置 app.jwt

```
router.get(`/api/v1`,app.jwt, controller.api.v1.index);  
router.get(`/api/v1/productList`,app.jwt, controller.api.v1.productList);
```

## 6、客户端请求

### Vue 中 get 请求

```
getIndex() {  
  var api = "http://localhost:7001/api/v1";  
  this.Axios.get(api, {  
    headers: {  
      Authorization: "Bearer " + this.token,  
    },  
  })  
  .then((response) => {  
    console.log(response.data);  
  })
```

```
.catch((err) => {  
    console.log(err);  
});  
},
```

## Vue 中 Post 请求

```
addPeopleInfo() {  
    var api = "http://localhost:7001/api/v1/addPeopleInfo";  
    this.Axios.post(api, {  
        tableId: 12,  
        pNum: 4,  
        pMark: "不要辣椒",  
    }, {  
        headers: {  
            Authorization: "Bearer " + this.token,  
        },  
    })  
    .then((response) => {  
        console.log(response.data);  
    })  
    .catch((err) => {  
        console.log(err);  
    });  
}
```

## Postman

