点餐系统

接口使用说明书

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [ ]正式发布  [√] 正在修改 | 文件标识： | 001 |
| 当前版本： | X.Y |
| 作 者： | 魏玉凤 |
| 完成日期： |  |

**目 录**

**[一、 接口概述 4](#_Toc12840)**

[1.1 概述 4](#_Toc23793)

[1.2 基本规则 4](#_Toc13221)

[1.3 域名 4](#_Toc15824)

[1.4 编写目的 4](#_Toc31278)

**[二、 接口定义 4](#_Toc25446)**

[1. IUserBLL接口 4](#_Toc32758)

[1.1. Register 注册方法 4](#_Toc18449)

[1.2. Login登录方法 5](#_Toc25358)

[1.3. MotifyPassword修改密码方法 5](#_Toc4636)

[1.4. GetAllUsers获取所有用户信息方法 5](#_Toc14096)

[2. IMenuBLL接口 6](#_Toc23377)

[2.1. AddDish增加菜品方法 6](#_Toc30949)

[1.2. DelDish删除菜品方法 6](#_Toc30223)

[1.3. MotifyDish修改菜品方法 6](#_Toc24078)

[1.4. FindAllDishs获取所有菜品信息方法 7](#_Toc9227)

[3. IOrderBLL接口 7](#_Toc17109)

[2.1. OrderFood订餐方法 7](#_Toc5160)

[1.2. CancelOrder取消订餐方法 7](#_Toc22343)

[1.3. ProduceSweep产生打扫人员方法 8](#_Toc26639)

[1.4. GetOrderPeople获取今日订餐人员信息方法 8](#_Toc26874)

[1.5. CountOrderNumber统计今日订餐人数方法 8](#_Toc18274)

[1.6. GetPersonalOrderInformation获取个人订餐信息方法 9](#_Toc22492)

## 接口概述

### 概述

接口用于点餐系统和外界系统进行数据交互以完成某种业务流程。

### **基本规则**

### **域名**

http://www.\*\*\*.com

### **编写目的**

预期读者：

对接口行为和目的有一定了解的人。

背景说明 

软件系统名称：接口前端接入服务器。 

描述 <接口> 接收不同商家的接入，接收数据转发给主机服务器，并同步/异步将 返回数据发给接入商家的行为。

## 接口定义

### 2.1 用户管理

#### 2.1.1 注册

1. 请求API

User/Register/{user}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| user | 用户的信息 |
| user.UserName | 用户名 |
| user.Password | 用户密码 |
| user.EmployeeId | 员工号 |
| user.UserType | 用户类型(0管理员，1普通用户) |

3. 请求参数示例

{

“UserName”:”zhangsan”,

“Password”:”123”,

“EmployeeId”:”NL1001”,

“UserType”:0

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 注册成功 |
| 500 | “系统繁忙” | 注册失败 |
| 404 | “两次密码不一致” | 注册失败 |

#### 2.1.2 登录

1. 请求API

User/Login/{userName}/{password}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| userName | 用户名 |
| password | 用户密码 |

3. 请求参数示例

{

“UserName”:”zhangsan”,

“Password”:”123”

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | UserInfo | 登录成功，返回User对象 |
| 404 | “用户名或密码错误” | 登录失败 |
| 500 | “系统繁忙” | 登录失败 |
| 202 | “用户名或密码为空” | 登录失败 |

#### 2.1.3 修改密码

1. 请求API

User/MotifyPassword/{userId}/{password}/{repeatPassword}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| userId | 用户编号 |
| password | 新密码 |
| repeatPassword | 重复输入的密码 |

3. 请求参数示例

{

“userId”:1,

“password”:”123”,

“repeatPassword”:”123”

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 修改成功 |
| 404 | “两次密码不一致” | 修改失败 |
| 500 | “系统繁忙” | 修改失败 |

#### 2.1.4 查询所有用户

1. 请求API

User/GetAllUsers

2. 请求参数描述

无

3. 请求参数示例

无

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | List<UserInfo> | 查询成功，返回List集合 |
| 500 | “系统繁忙” | 查询失败 |

#### 2.1.5 验证用户名是否重复

1. 请求API

User/UserNameIsRepeat/{userName}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| userName | 用户名 |

3. 请求参数示例

{

“userName”:”zs”

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “该用户名可以使用” | 该用户名可以使用 |
| 404 | “该用户名已存在” | 数据库中存在该用户 |
| 500 | “系统繁忙” | 查询失败 |

### 2.2 菜单管理

#### 2.2.1 增加菜单

1. 请求API

Menu/AddDish/{menu}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| menu | 菜单 |
| menu.MenuName | 菜品名称 |
| menu.Price | 菜品价格 |
| menu.MenuType | 菜品类型（0荤菜1素菜2冷菜） |

3. 请求参数示例

{

“MenuName”:”糖醋排骨”,

“Price”:”30”,

“MenuType”:0

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 菜品增加成功 |
| 500 | “系统繁忙” | 菜品增加失败 |
| 404 | “菜名为空” | 菜品增加失败 |

#### 2.2.2 修改菜单

1. 请求API

Menu/MotifyDish/{menu}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| menu | 菜单 |
| menu.MenuName | 菜品名称 |
| menu.Price | 菜品价格 |
| menu.MenuType | 菜品类型（0荤菜1素菜2冷菜） |

3. 请求参数示例

{

“MenuName”:”糖醋里脊”,

“Price”:”32”,

“MenuType”:0

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 菜品修改成功 |
| 500 | “系统繁忙” | 菜品修改失败 |

#### 2.2.3 删除菜单

1. 请求API

Menu/DelDish/{menu}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| menuId | 菜单编号 |

3. 请求参数示例

{

“menuId”:1

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 菜品删除成功 |
| 500 | “系统繁忙” | 菜品删除失败 |

#### 2.2.4 查询全部菜单

1. 请求API

Menu/FindMenu

2. 请求参数描述

无

3. 请求参数示例

无

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | List<MenuInfo> | 菜单查询成功，返回List集合 |
| 500 | “系统繁忙” | 菜单查询失败 |

#### 2.2.5 上传图片

1. 请求API

Menu/UploadPic

2. 请求参数描述

3. 请求参数示例

4. 返回状态码说明

### 2.3 订餐管理

#### 2.2.1 订餐

1. 请求API

Order/OrderFood/{userId}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| userId | 用户编号 |

3. 请求参数示例

{

“userId”:1

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 订餐成功 |
| 500 | “系统繁忙” | 订餐失败 |

#### 2.2.2 取消订餐

1. 请求API

Order/CancelOrder/{userId}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| userId | 用户编号 |

3. 请求参数示例

{

“userId”:1

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 取消订餐成功 |
| 500 | “系统繁忙” | 取消订餐失败 |

#### 2.2.3 产生打扫人员

1. 请求API

Order/ProduceSweep

2. 请求参数描述

无

3. 请求参数示例

无

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 打扫人员产生成功 |
| 500 | “系统繁忙” | 打扫人员产生失败 |

#### 2.2.4 获取今日订餐人员信息

1. 请求API

Order/GetOrderPeople

2. 请求参数描述

无

3. 请求参数示例

无

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | “OK” | 查询成功 |
| 500 | “系统繁忙” | 查询失败 |

#### 2.2.5 统计今日订餐人数

1. 请求API

Order/ConutOrderNumber

2. 请求参数描述

无

3. 请求参数示例

无

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | number | 统计成功，返回int类型 |
| 500 | “系统繁忙” | 统计失败 |

#### 2.2.2 获取个人订餐信息

1. 请求API

Order/GetPersonalOrderInformation/{userId}

2. 请求参数描述

|  |  |
| --- | --- |
| **参数** | **说明** |
| userId | 用户编号 |

3. 请求参数示例

{

“userId”:1

}

4. 返回状态码说明

|  |  |  |
| --- | --- | --- |
| **状态码** | **结果** | **描述** |
| 200 | List<OrderInfo> | 查询成功，返回List集合 |
| 500 | “系统繁忙” | 查询失败 |