**客户名字和logo**

餐厅订餐系统-概要设计说明书

LUO ZHI

日期：2015-01-10

文档版本历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本号 | 修订日期 | 修订人 | 审核人 | 变更内容 |
| 0.1 | 2015-01-08 | LUO ZHI |  | 初始化 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1 文档介绍 1](#_Toc408510009)

[1.1 文档目的 1](#_Toc408510010)

[1.2 文档范围 1](#_Toc408510011)

[1.3 缩写词列表 1](#_Toc408510012)

[1.4 参考内容 1](#_Toc408510013)

[2 系统架构图 2](#_Toc408510014)

[3 系统模块图 3](#_Toc408510015)

[3.1 权限管理模块 3](#_Toc408510016)

[3.1.1 模块概述 3](#_Toc408510017)

[3.1.2 功能划分 3](#_Toc408510018)

[3.1.3 使用技术 4](#_Toc408510019)

[3.2 移动点菜模块 4](#_Toc408510020)

[3.2.1 模块概述 4](#_Toc408510021)

[3.2.2 功能划分 4](#_Toc408510022)

[3.2.3 使用技术 4](#_Toc408510023)

[4 系统模块流程/关系？？ 4](#_Toc408510024)

[5 系统接口实现技术 4](#_Toc408510025)

[5.1 REST风格的Web-Service 4](#_Toc408510026)

[5.2 JSON数据格式 5](#_Toc408510027)

[5.3 安全性 5](#_Toc408510028)

[5.4 接口协议 5](#_Toc408510029)

[5.4.1 获取用户信息 5](#_Toc408510030)

[5.4.2 获取目前所有分类列表 6](#_Toc408510031)

[5.4.3 获取某一个分类下面所有菜品列表 7](#_Toc408510032)

[5.4.4 查询某个会员的历史订餐记录 8](#_Toc408510033)

[5.4.5 查询某个订单详情 8](#_Toc408510034)

[6 数据库设计 10](#_Toc408510035)

[6.1 数据字典 10](#_Toc408510036)

# 文档介绍

## 文档目的

文档基于架构设计说明书与需求规格说明书，对系统模块及其子功能进行定义，对系统数据字典进行描述，对Web-Service接口协议进行规范，开发团队基于本文档对系统模块功能的具体实现进行详细设计文档编写。

## 文档范围

本文的仅对系统模块功能定义，使用技术，接口协议与数据字典进行描述，系统架构，需求以及具体实现均不在本文档范围内。

## 缩写词列表

|  |  |
| --- | --- |
| 缩写词 | 解释 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

## 参考内容

# 系统架构图

系统机构和技术方案在*系统架构说明书*中有详细阐述，根据系统架构说明书总结系统开发技术方案如下表，

|  |  |
| --- | --- |
| **MVC** | Hibernate 3.x |
| Spring 3.x |
| Struts2 |
| **WEB** | Bootstrap 3 |
| JQuery |
| **移动应用方案** | Android 4.x |
| **Web-Service** |  |
| **数据库** |  |

# 系统模块图

根据需求规格说明书中的需求定义，餐厅订餐系统可以分为移动点餐模块，后台收银模块，菜单管理模块，会员管理模块和用户权限管理模块共五个模块，系统模块图下图所示



## 权限管理模块

### 模块概述

权限管理模块用于维护系统用户，系统权限，系统角色的数据以及三者之间的关系。系统权限唯一映射了对系统某一功能的使用权力，例如添加用户这个功能；那么系统角色则定义了一类有相关性或者依赖性的功能集合，例如管理员角色能使用权限管理模块中的所有功能；而系统用户是系统角色的拥有者，通过角色获得系统功能使用的权利。

### 功能划分

* **用户登录：**用户使用用户名和与用户名匹配的密码登录到系统，根据用户拥有的角色获得不同的系统菜单；使用无效用户名或者错误的密码，会得到系统的错误信息提示
* **创建/编辑用户信息：**拥有权限管理模块功能使用权限的系统用户，可以使用这个功能创建新的用户和更新现有用户信息
* **用户角色分配：**拥有权限管理模块功能使用权限的系统用户，使用这个功能为系统用户分配相应的角色

### 使用技术

* Hibernate
* Struts2
* Spring
* JSP with Tiles
* Bootstrap
* JQuery

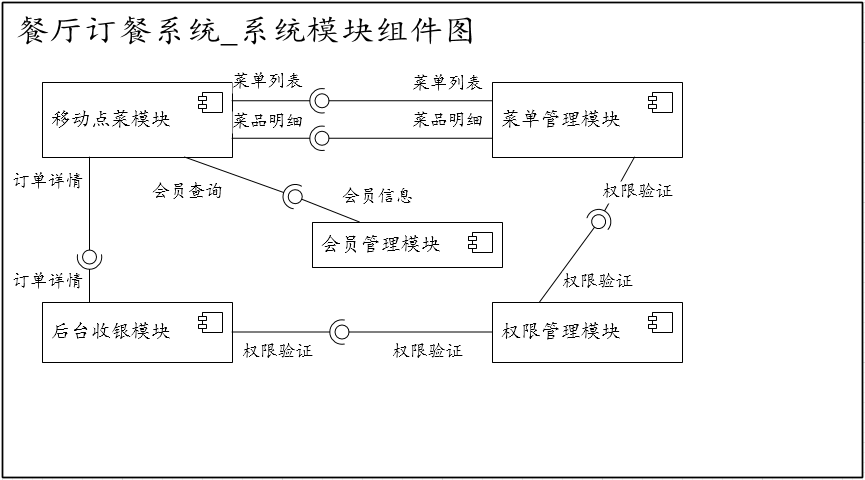
## 移动点菜模块

### 模块概述

### 功能划分

### 使用技术

# 系统模块流程/关系



# 系统接口实现技术

餐厅订餐系统五大模块之间没有明显的互相调用，所以模块与模块之间没有明显的接口定义。由其他模块输入的数据，均有使用模块直接访问数据库获取。

餐厅订餐系统的移动点菜模块物理上分为移动客户端和后台Web-Service服务，移动客户端通过消费后台Web-Service服务完成点菜流程。后台Web-Service使用REST风格定义，并使用JSON作为数据格式。

## REST风格的Web-Service

REST风格中文直译是含状态传输（Representational State Transfer），与传统的SOAP和XML-RPC等方式相比更加简洁与容易理解。REST风格扩充使用了HTTP协议中不常被使用的PUT和DELETE这2个动词（相对于GET和POST更为大家熟悉而言）。

REST风格使用URI来定义资源，每一个资源都有唯一的标识符；使用HTTP协议提供的GET/POST/PUT/DELETE四种方法来定义对资源的动作。恰好这四种方法又可以映射到数据库操作的CRUD四种基本操作。下表是一个简单的REST风格举例，

|  |  |  |  |
| --- | --- | --- | --- |
| 资源 | | User | |
| 动作 | GET | <http://host/ws/users> | 获取所有用户集合 |
| GET | <http://host/ws/users/1> | 获取ID为1的用户 |
| POST | <http://host/ws/users> | 创建一个新用户 |
| PUT | <http://host/ws/users/1> | 更新ID为1的用户数据记录 |
| DELETE | <http://host/ws/users/1> | 删除ID为1的用户数据记录 |

## JSON数据格式

JSON是一种类似于XML的数据表示格式，但是比XML更为简洁和轻量，它没有XML众多的Element节点，也不需要进行Validation，尤其JSON与JavaScript的天然集成，有效提高了开发效率。

在JSON的世界里，只有对象。JSON使用花括弧{}和键值对来定义对象，下面是一个使用JSON来表示用户对象的一个简单例子，

*{*

*“username”: “Jerry Gao”,*

*“password” : “12345”,*

*“mobile\_phone”: “13312345678”*

*“roles”: [*

*{*

*“role\_name”: “admin”*

*},*

*{*

*“role\_name”: “cashier”*

*}*

*]*

*}*

## 安全性

使用HTTPS加密通道以及BASE64加密算法对数据进行保护。

## 接口协议

### 获取用户信息

* URL：http://192.168.0.100:8080/eorder-ws/rest/users/myuser/13988888888
* 方法：GET
* 输入：手机号码

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 参数类型 | 说明 | 备注 |
| cellphone | String | 手机号码 | 例如：13988888888 |

* 输出： 会员信息

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 参数类型 | 说明 | 备注 |
| userId | int | 分类id | 例如： 1 |
| userName | String | 分类名称 | 例如：张三 |
| cellphone | String | 该分类的图片 | 例如：13988888888 |
| levelName | String | 用户等级名称 | 金牌会员 |
| discount | Float | 折扣信息 | 例如：钻石8折、白金8.5折、金牌9折、普通会员9.5折等等。 |

* 样列：

{

"user":

{

"userId": 1,

"userName": "张三",

"cellphone": “13988888888”,

"levelName": “金牌会员”,

"discount：9

}

}

### 获取目前所有分类列表

* URL：http://192.168.0.100:8080/eorder-ws/rest/categories
* 方法：GET
* 输入：无
* 输出：分类列表，包含分类名，分类id字段，分类图片URL

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 参数类型 | 说明 | 备注 |
| categoryId | int | 分类id | 例如： 1 |
| categoryName | String | 分类名称 | 例如：最新 |
| categoryPicture | String | 该分类的图片 | 例如：http://www....33/21.jpg |

* 举例：

{

"categories": [

{

"categoryId": 1,

"categoryName": "最热",

"categoryPicture": null

},

{

"categoryId": 2,

"categoryName": "最新",

"categoryPicture": null

},

{

"categoryId": 3,

"categoryName": "湘菜",

"categoryPicture": null

}

]

}

### 获取某一个分类下面所有菜品列表

* URL：http://localhost:8080/eorder-ws/rest/dishes/mydishes/6
* 方法：GET
* 输入：分类id

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 名称 | 1. 参数类型 | 1. 说明 | 1. 备注 |
| 1. categoryId | 1. int | 1. 分类id | 1. 例如： 1 |

* 输出：菜品列表，包含菜品名、菜品id字段、菜品价格和菜品图片URL

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 名称 | 1. 参数类型 | 1. 说明 | 1. 备注 |
| 1. dishId | 1. int | 1. 菜品id | 1. 例如：30001 |
| 1. dishName | 1. String | 1. 菜品名称 | 1. 例如：麻婆豆腐 |
| 1. dishPicture | 1. String | 1. 该菜品的图片 | 1. 例如：http://www....33/31.jpg |
| 1. dishPrice | 1. Float | 1. 该菜品的价格 | 1. 例如：48.5 |

* 举例：

｛dishes:[

｛＂dishId＂：＂30001＂，＂dishName＂：＂麻婆豆腐＂，＂dishPrice＂：＂18＂，＂dishPicture＂：＂http://www.google.com/2332333/31.jpg＂｝

｛＂dishId＂：＂30002＂，＂dishName＂：＂豆花鱼＂，＂dishPrice＂：＂38＂，＂dishPicture＂：＂http://www.google.com/2332333/32.jpg＂｝

｛＂dishId＂：＂30003＂，＂dishName＂：＂夫妻肺片＂，＂dishPrice＂：＂28.9＂，＂dishPicture＂：＂http://www.google.com/2332333/33.jpg＂｝]

.......

}

### 查询某个会员的历史订餐记录

* URL：http://localhost:8080/eorder-ws/rest/orders/myorders/13912345673
* 方法：GET
* 输入：会员手机号码

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 名称 | 1. 参数类型 | 1. 说明 | 1. 备注 |
| 1. cellphone | 1. String | 1. 手机号码 | 1. 例如： 13988888888 |

* 输出：会员历史订单记录列表, 包含订单id,　订单日期，总价;

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 名称 | 1. 参数类型 | 1. 说明 | 1. 备注 |
| 1. orderId | 1. int | 1. 订单id | 1. 例如： 1 |
| 1. createAt | 1. String | 1. 订单时间 | 1. 例如：2014-8-1 |
| 1. totalPrice | 1. Float | 1. 订单的总价 | 1. 例如：104.5 |

* 举例：

｛”orders”:

[ ｛＂orderId＂：＂40001＂，＂createAt＂：＂2014-8-1＂,＂totalPrice＂： "104.5"｝

｛＂orderId＂：＂40002＂，＂createAt＂：＂2014-9-15＂, ＂totalPrice＂："234.5"｝

｛＂orderId＂：＂40003＂，＂createAt＂：＂2014-10-11＂, ＂totalPrice＂："800"｝]

.......

｝

### 查询某个订单详情

* URL：http://localhost:8080/eorder-ws/rest/orderitems/myorderitems/{orderId}
* 方法：GET
* 输入：订单id

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 名称 | 1. 参数类型 | 1. 说明 | 1. 备注 |
| 1. orderId | 1. int | 1. 菜品id | 1. 例如： 1 |

* 输出：订单详情，包括订单号，订单日期，订单总价，订单菜品列表，单项菜品数量;

|  |  |  |  |
| --- | --- | --- | --- |
| 1. 名称 | 1. 参数类型 | 1. 说明 | 1. 备注 |
| 1. dishId | 1. int | 1. 菜品id | 1. 例如： 1 |
| 1. dishName | 1. String | 1. 菜品名称 | 1. 例如：麻婆豆腐 |
| 1. dishPrice | 1. Float | 1. 菜品的单价 | 1. 例如：18 |
| 1. dishAmount | 1. int | 1. 菜品的数量 | 1. 例如：１ |
| 1. dishPicture | 1. String | 1. 菜品的图片 | 1. 例如：http://www....33/31.jpg |

* 举例：

｛ “orderitems”：[

｛＂dishId＂：＂30001＂，＂dishName＂：＂麻婆豆腐＂，＂dishPrice＂：＂18＂，　＂dishAmount＂：＂１＂，＂dishPicture＂：＂http://www.google.com/2332/31.jpg＂｝

｛＂dishId＂：＂60001＂，＂dishName＂：＂鱼香肉丝＂，＂dishPrice＂：＂32＂，＂dishAmount＂：＂１＂，＂dishPicture＂：＂http://www.google.com/2332/88.jpg＂｝

｛＂dishId＂：＂80001＂，＂dishName＂：＂紫菜蛋汤＂，＂dishPrice＂：＂28＂，＂dishAmount＂：＂１＂，＂dishPicture＂：＂http://www.google.com/2333/92.jpg＂｝

]｝

# 数据库设计

## 数据字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 菜品分类表 | | t\_dish\_category | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 菜品类别ID | category\_id | 整型 | 32 | n/a | 主键，自动增加 |
| 类别名称 | name | 字符型 | 12 | n/a | 非空 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 菜品表 | | t\_dish | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 菜品类别ID | dish\_id | 整型 | 32 | n/a | 主键，自动增加 |
| 菜品分类ID | category\_id | 整型 | 32 | 0 | 非空 |
| 菜品名称 | name | 字符型 | 12 | n/a | 非空 |
| 价格 | price | 浮点型 | 32 | 0.0 | 精确到小数点后一位 |
| 是否在售 | on\_sell | 布尔型 | 1 | True | 非空 |
| 其他信息 | misc | 字符串 | 128 | n/a | n/a |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |
| 数据表名 | 订单表 | | t\_order | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 订单ID | order\_id | 整型 | 32 | n/a | 主键，自动增加 |
| 订单流水号 | order\_seq | 字符型 | 16 | n/a | 格式为<yyyymmddxxxxx>  yyyy四位数年份  mm两位数月份  dd两位日期  xxxxx当天订单序号 |
| 订单状态 | order\_status | 字符型 | 16 | ‘new’ | 非空 |
| 台号 | table\_num | 整型 | 16 | n/a |  |
| 人数 | attendee\_num | 整型 | 16 | 1 |  |
| 实收金额 | total\_price | 浮点型 | 32 | 0.0 | 精确小数点后一位 |
| 服务员ID | servent\_id | 整型 | 32 | n/a | 非空 |
| 会员ID | member\_id | 整型 | 32 | n/a | n/a |
| 收银员ID | casher\_id | 整型 | 32 | n/a | n/a |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 订单条目表 | | t\_order\_item | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 订单条目ID | item\_id | 整型 | 32 | n/a | 主键，自动增加 |
| 订单ID | order\_id | 整型 | 32 | 0 |  |
| 菜品ID | dish\_id | 整型 | 32 | 0 | 非空 |
| 份数 | dish\_acc | 整型 | 32 | 1 | 最大99 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 用户表 | | t\_user | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 用户ID | user\_id | 整型 | 32 | n/a | 主键，自动增加 |
| 用户名 | username | 字符型 | 16 | n/a | 非空，英文字母和数字组成不少于6位 |
| 密码 | password | 字符型 | 128 | n/a | 非空，英文字母和数字组成不少于6位 |
| 手机号 | cellphone | 字符型 | 16 |  |  |
| 会员等级ID | level\_id | 整型 | 32 |  |  |
| 用户状态 | user\_status | 布尔类型 | 1 | n/a | 非空 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 会员等级表 | | t\_user\_level | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 会员等级ID | level\_id | 整型 | 32 | n/a | 主键，自动增加 |
| 等级名称 | level\_name | 字符型 | 16 | n/a |  |
| 等级分数 | level\_score | 整型 | 32 | n/a |  |
| 等级状态 | level\_status | 布尔类型 | 1 | n/a | 非空 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 角色表 | | t\_role | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 角色ID | role\_id | 整型 | 32 | n/a | 主键 |
| 角色名称 | role\_name | 字符串 | 32 | n/a | 非空 |
| 角色描述 | role\_desc | 字符串 | 128 | n/a | 非空 |
| 角色状态 | role\_status | 布尔类型 | 1 | n/a | 非空 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 用户角色关联表 | | t\_user\_role | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 用户ID | user\_id | 整型 | 32 | n/a | 主键 |
| 角色ID | role\_id | 整型 | 32 | n/a | 主键 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 功能表 | | t\_function | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 功能ID | function\_id | 整型 | 32 | n/a | 主键 |
| 功能名称 | function\_name | 字符串 | 32 | n/a | 非空 |
| 功能描述 | function\_desc | 字符串 | 128 | n/a | 非空 |
| 功能路径 | function\_path | 字符串 | 128 | n/a | 非空 |
| 父功能ID | function\_parent | 整型 | 32 | n/a | 可为空 |
| 功能排序 | function\_order | 字符串 | 32 | n/a | 非空 |
| 功能状态 | function\_status | 布尔类型 | 1 | n/a | 非空 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据表名 | 角色功能关联表 | | t\_role\_function | | |
| 数据项描述 | 数据项名 | 类型 | 长度 | 默认值 | 限制 |
| 功能ID | function\_id | 整型 | 32 | n/a | 主键 |
| 角色ID | role\_id | 整型 | 32 | n/a | 主键 |
| 创建时间 | create\_at | 日期型 |  | 当前日期 | 非空 |
| 修改时间 | update\_at | 日期型 |  | 当前日期 | 非空 |