目录

[一、通用模块-common 1](#_Toc483298346)

[1.1 功能简述 1](#_Toc483298347)

[1.0.1 通用部分 1](#_Toc483298348)

[1.1.1 注册账户 3](#_Toc483298349)

[1.1.2 注销账户 5](#_Toc483298350)

[1.1.3 修改账户信息 6](#_Toc483298351)

[1.1.4 查询账户信息 8](#_Toc483298352)

[1.1.5 登录验证 9](#_Toc483298353)

[1.1.6 登录信息记录 10](#_Toc483298354)

[1.1.7 账户信息修改记录 11](#_Toc483298355)

[1.1.8 数据反馈 12](#_Toc483298356)

[1.1.9 验证码 13](#_Toc483298357)

[1.2 数据库设计 13](#_Toc483298358)

[1.2.1 数据库设计思路 13](#_Toc483298359)

[1.2.2 数据库架构图 14](#_Toc483298360)

[1.2.3 数据库详细说明 15](#_Toc483298361)

[附录A 状态码 16](#_Toc483298362)

[模块标识码 16](#_Toc483298363)

[功能标识码 16](#_Toc483298364)

[特殊返回值 16](#_Toc483298365)

# 一、通用模块-common

## 1.1 功能简述

该系统采用的是RBAC（role-based access control，基于角色的访问控制），因此账户模块被融入到通用模块之中(不包含商家账户)。

### 1.1.0 通用部分

该部分主要包含的是其他几个模块所使用的一些通用部分，以及封装返回值信息设计，其主要包含：

* 检查账号的用户名是否已被注册
* 检查手机是否已被注册
* 检查邮箱是否已被注册
* 检查账号的登录状态

注意：检查账号的登录状态不涉及到数据库层。

#### 返回信息格式(JSON)

{"code":"0","message":"success"} //code状态码，message信息

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000000 | 手机号未被注册 | 空 |
| 000001 | 手机号已被注册 |
| 000002 | 验证的手机号格式异常 |
| 000003 | 账号已被注册 |
| 000004 | 账号未被注册 |
| 000005 | 邮箱已被注册 |
| 000006 | 邮箱未被注册 |
| 00007 | 验证的邮箱格式异常 |
| 000008 | 登录状态：未登录 |
| 000009 | 登录状态：已登录 |
| 000010 | 邮箱未被激活 |
| 000011 | 邮箱已被激活 |
| 000012 | 未激活邮箱清理异常 |
| 000013 | 激活信息写入异常 |
| 000014 | 激活信息写入成功 |
| 000015 | 账号为手机号或邮箱 |

### 1.1.1 注册账户

该模块针对底层实现来说是对账户的添加操作，对使用者而言就是注册操作。

**注册必填信息包括**：

* 密码

**必须包含信息（至少包含其一）：**

* 邮箱
* 手机号

**注册选填信息包括**：

* 账户名
* 性别
* 头像图片

**系统生成信息包括**：

* 账户ID(UUID)
* 注册时间
* 账户状态

注意事项：邮箱没有默认值，而头像图片有指定的默认值。手机号是固定长度的11位数字，注册时间是禁止修改的。

无法注册情况：

* 账户名已存在
* 账户名为手机号
* 账户名为邮箱
* 手机号格式不正确
* 邮箱地址不正确
* 账户名为手机号
* 账户名为邮箱地址

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000100 | 账户注册成功 | AccountSimple |
| 000101 | 账户已存在，注册失败 | NULL |
| 000102 | 账户名为手机号注册失败 |
| 000103 | 账户名为邮箱，注册失败 |
| 000104 | 系统异常导致账户注册失败 |
| 000105 | 缺少必要信息，注册失败 |
| 000106 | 信息不符合格式，注册失败 |
| 000107 | 手机号已被注册 |
| 000108 | 邮箱已被注册 |
| 000109 | 密码两次输入不一致 |
| 000110 | 验证码未通过，注册失败 |

### 1.1.2 注销账户

该模块针对底层实现来说是对账户的删除操作，对使用者而言就是注销操作。

删除操作提供三类操作：

* 删除权限：当某种权限不再需要的时候
* 删除角色：当某种角色不再需要的时候，或者说用户的某个角色注销的时候
* 删除账户：单独某个账户的清除，包含其权限和角色信息。可通过账户的ID，账户名称两种方式进行删除操作。

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

{"code":"0","message":"success","data":[]} // data 为 array 的情况

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000200 | 账户注销成功 |  |
| 000201 | 账户注销失败：账户不存在 |  |
| 000202 | 账户注销失败：账户未注销 |  |

账户移除成功

账户的角色移除成功

权限不足，无法移除账户的角色

权限不足，无法删除账户

该类角色移除成功(保留账户)

该类角色的账户移除成功(保留角色)

该类角色及其账户移除成功

系统异常导致账户移除失败

系统异常导致账户的角色移除失败

系统异常导致导致该类角色移除失败

系统异常导致该类角色的账户移除失败

系统异常导致该类角色的账户移除失败

### 1.1.3 修改账户信息

该模块针对底层实现来说是对账户的更新操作，对使用者而言就是修改个人信息操作。

**可修改的个人信息包括**：

* 密码
* 性别
* 手机号
* 邮箱
* 头像图片
* 账号的状态
* 角色

**禁止修改的个人信息包括**：

* 账户ID、
* 账户名称
* 注册时间

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

{"code":"0","message":"success","data":[]} // data 为 array 的情况

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000300 | 修改失败:账户ID不存在 | null |
| 000301 | 账户密码修改成功 | Account |
| 000302 | 修改失败:密码未修改 | null |
| 000303 | 修改失败:密码修改异常 | null |
| 000304 | 账号性别修改成功 | Account |
| 000305 | 修改失败:性别未修改 | null |
| 000306 | 修改失败:性别修改异常 | null |
| 000307 | 账号手机号修改成功 | Account |
| 000308 | 修改失败:手机号未修改 |  |
| 000309 | 修改失败:手机号修改异常 |  |
| 000310 | 修改失败：手机号格式异常 |  |
| 000311 | 账号状态修改成功 | Account |
| 000312 | 修改失败:账号状态未修改 |  |
| 000313 | 修改失败:账号状态修改异常 |  |
| 000314 | 账号邮箱修改成功 |  |
| 000315 | 修改失败:账号邮箱未修改 |  |
| 000316 | 修改失败:账号邮箱修改异常 |  |
| 000317 | 修改失败:该账号不存在 |  |
| 000318 | 修改失败:修改包含禁止修改信息 |  |
| 000319 | 修改失败:账号信息未修改 |  |
| 000320 | 修改失败:账号信息修改异常 |  |
| 000321 | 修改失败:账号不存在 |  |
| 000322 | 账号信息修改成功 |  |
| 000323 | 修改失败：邮箱格式异常 |  |
| 000324 | 修改密码失败:输入原密码错误 |  |
| 000325 | 修改密码失败：两次输入新密码错误 |  |
| 000326 | 修改密码失败：未通过验证码验证 |  |
| 000327 | 修改手机号失败：未通过验证码验证 |  |
| 000328 | 修改失败:用户名修改异常 |  |
| 000329 | 修改失败：用户名未修改 |  |
| 000330 | 账号用户名修改成功 | Account |
| 000331 | 修改失败：头像未修改 |  |
| 000332 | 修改失败：头像修改异常 |  |
| 000333 | 账号头像修改成功 | Account |

### 1.1.4 查询账户信息

该模块针对底层实现来说是对账户的查询操作，对使用者而言就是浏览账户信息操作。该模块是进行账户查询操作，返回的是账户的信息。

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

{"code":"0","message":"success","data":[]} // data 为 array 的情况

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000400 |  |  |
| 000401 |  |  |

### 1.1.5 登录验证

该模块针对底层实现来说是对账户的查询操作，对使用者而言就是登录账户验证操作。只不过这个操作是一种比较常用的查询账户信息操作，因此将其单独进行设计，与上面的查询账户信息模块不同之处是：该模块只针对单独账户，返回的只是该角色的权限及基本信息。

验证方式可以通过手机号、邮箱、账户这三种方式进行登录验证，而用户无须手动选择，只需要填写登录信息即可，这部分是由后台进行区分并处理的。

如果登录验证成功，会向t\_logininfo插入一条登录信息

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

{"code":"0","message":"success","data":[]} // data 为 array 的情况

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000500 | 账号验证成功 | AccountSimple |
| 000501 | 账户验证失败：账号不存在 | null |
| 000502 | 账户验证失败：密码错误 |
| 000503 | 系统异常导致账号验证失败 |
| 000504 | 账号验证成功，但是登录信息保存失败 | AccountSimple |
| 000505 | 登录手机号异常 | null |
| 000506 | 验证码未通过，登录失败 |
| 000507 | 账号未登录，无需注销登录 |
| 000508 | 账号注销登录成功 |
| 000509 | 账号处于未激活状态，登陆失败 |
| 000510 | 邮箱未激活，无法进行登陆 |

### 1.1.6 登录信息记录

该模块主要功能是记录用户所有登录操作，无论登录账号是否存在，密码是否正确。该表的主要目的：

* 筛选出长期不登录账号
* 检测异常登录IP及设备

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

{"code":"0","message":"success","data":[]} // data 为 array 的情况

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000600 | 账号密码匹配正确：登录信息记录成功 |  |
| 000601 | 账号存在：密码错误，登录信息记录成功 |  |
| 000602 | 账号不存在：登录信息记录成功 |  |
| 000603 | 系统异常导致登录信息记录失败 |  |

### 1.1.7 账户信息修改记录

该模块主要功能是记录账户信息的修改等。由于账户可能被删除，但是这些记录信息是不可以消除的，所以数据库底层将他们并没有直接关联，但是他们确是高度同步的。

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

{"code":"0","message":"success","data":[]} // data 为 array 的情况

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000700 | 密码修改成功:账户修改信息记录成功 |  |
| 000701 | 性别修改成功:账户修改信息记录成功 |  |
| 000702 | 手机号修改成功:账户修改信息记录成功 |  |
| 000703 | 邮箱修改成功:账户修改信息记录成功 |  |
| 000704 | 头像修改该成功:账户修改信息记录成功 |  |
| 000705 | 账号状态修改成功:账户修改信息记录成功 |  |
| 000706 | 角色修改成功:账户修改信息记录成功 |  |
| 000707 | 修改禁止修改信息：账户修改信息记录成功 |  |
| 000708 | 系统异常导致账户修改信息记录失败 |  |

### 1.1.8 数据反馈

主要是反馈数据信息。主要功能包含：

* 反馈省市县三级地址数据
* 通过地址筛选出当前地址下的商家
* 通过地址筛选出当前地址下的商家，以及根据距离进行排序
* 通过地址筛选出当前地址下的商家，以及根据价格进行排序

#### 返回信息格式(JSON)

{"code":"0","message":"success","data":{}} //code状态码，message信息，data返回数据

{"code":"0","message":"success","data":[]} // data 为 array 的情况

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000800 | 店铺已被已收藏 |  |
| 000801 | 店铺未被收藏 |  |
| 000802 | 店铺不存在 |  |
| 000803 | 账户不存在 |  |
| 000804 | 店铺收藏成功 |  |
| 000805 | 店铺收藏失败 |  |
| 000806 | 取消店铺收藏成功 |  |
| 000807 | 取消店铺收藏失败 |  |
| 000808 | 评论失败 |  |
| 000809 | 评论成功 |  |

### 1.1.9 验证码

针对手机验证码服务，其主要包括：

* 请求发送验证码
* 验证码输入验证

针对邮箱验证服务，其主要包括：

* 请求发送验证连接
* 验证连接验证

#### 返回信息格式(JSON)

{"code":"…","message":"…" } //code状态码，message信息

#### 状态码及对应信息

|  |  |  |
| --- | --- | --- |
| **状态码** | **信息** | **返回data格式** |
| 000900 | 验证码发送成功 | 空 |
| 000901 | 验证码发送失败：短信账户余额不足 |
| 000902 | 验证码发送失败：手机号格式不正确 |
| 000903 | 验证码发送失败：手机号不存在 |
| 000904 | 验证码发送失败：开发者账户异常 |
| 000905 | 验证码发送失败：非预期异常 |
| 000906 | 验证码发送失败：验证码格式不正确 |
| 000907 | 验证码错误 |
| 000908 | 验证码已过期 |
| 000909 | 接收到验证码为空，无法验证 |
| 000910 | 验证码验证成功 |
| 000911 | 邮箱验证链接发送成功 |
| 000912 | 邮箱验证链接发送失败 |
| 000913 | 邮箱验证链接错误 |
| 000914 | 邮箱验证链接失效 |
| 000915 | 邮箱验证链接验证成功 |

## 1.2 数据库设计

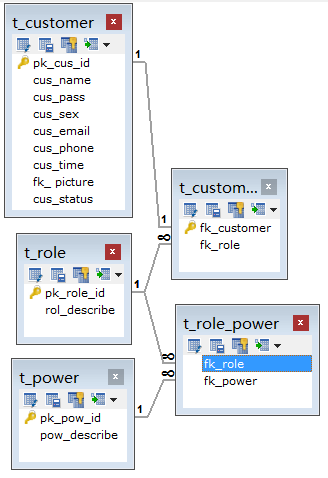
### 1.2.1 数据库设计思路

系统采用的是RBAC（role-based access control，基于角色的访问控制）权限管理，每个人可以有多个角色，每个角色又以有多种权限，反过来同一个权限可以对应多个角色，同一个角色可以对应多个人，因此这三者是多对多的关系，因此需要采用中间表进行设计。

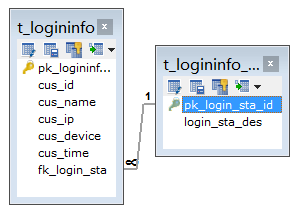
登录是最常用的操作，对登录信息进行记录也是必要的。设计初衷是检测异常登录，以及剔除长期未登录用户。

### 1.2.2 数据库架构图

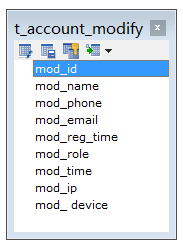
#### <1> RBAC结构



#### <2> 登录信息记录



#### <3> 账户信息修改记录



### 1.2.3 数据库详细说明

t\_customer为账户表，保存的是账户的基本信息；t\_role为角色表，保存的只是该系统包含的角色；t\_power，保存的是账户的权限。

t\_custome\_role是账户与角色的关联表；t\_role\_power是角色与权限的关联表。

t\_logininfo与t\_account\_modify是信息记录表，主要是记录用户的登录信息，以及账户修改信息记录。

# 附录A 状态码

状态码采用六位数字构成，其由三部分组成，每部分又由两个数字构成。

其构成格式为：模块标识码+功能标识码+信息标识码

注：信息标识码可详见各模块的功能说明。

## 模块标识码

|  |  |
| --- | --- |
| **模块** | **标识码** |
| common | 00 |

## 功能标识码

|  |  |
| --- | --- |
| **模块** | **标识码** |
| 通用部分 | 00 |
| 注册账户 | 01 |
| 注销账户 | 02 |
| 修改账户信息 | 03 |
| 查询账户信息 | 04 |
| 登录验证 | 05 |
| 登录信息记录 | 06 |
| 账户信息修改记录 | 07 |
| 数据反馈 | 08 |
| 验证码 | 09 |

## 特殊返回值

|  |  |
| --- | --- |
| **状态码** | **信息** |
| **xxxxxx** | **请求接口不存在，请重新确认** |
| **xxxx00** | **请求参数异常** |
| **xxxx01** | **请求必要参数审查通过** |
| **xxxx03** | **请求顺序错误，导致此次请求无法执行** |
| **xxxx04** | **请求所需必要元素已包含，可以继续执行** |

注意：如果返回红色标注的状态码，标识后台代码异常，请及时联系后台开发人员。