

天津大学

119 商城的设计文档



学 院	智能与计算学部
专 业	软件工程
年 级	2018级
姓 名	李昭卓
学 号	3018216132
姓 名	吕鸿江
学 号	3018216134
姓 名	康子文
学 号	3018216131
姓 名	马力
学 号	3018216137

2020 年 11 月 10 日

目录

设计文档.....	3
1 业务模型设计.....	3
2 接口设计.....	4
2.1 用户/user	4
2.2 商品/goods	8
2.3 评论/comment.....	10
2.4 购物车/cart.....	11
2.5 订单/order	13
2.6 账户/account	14
3 数据库设计.....	15
3.1E-R 图设计	15
3.2 数据表展示.....	16

设计文档

设计文档将从三个方面来描述：数据库设计，接口设计，业务模型设计三个方面来描述：

1 业务模型设计

本项目拟采用 MVC 三层架构进行书写

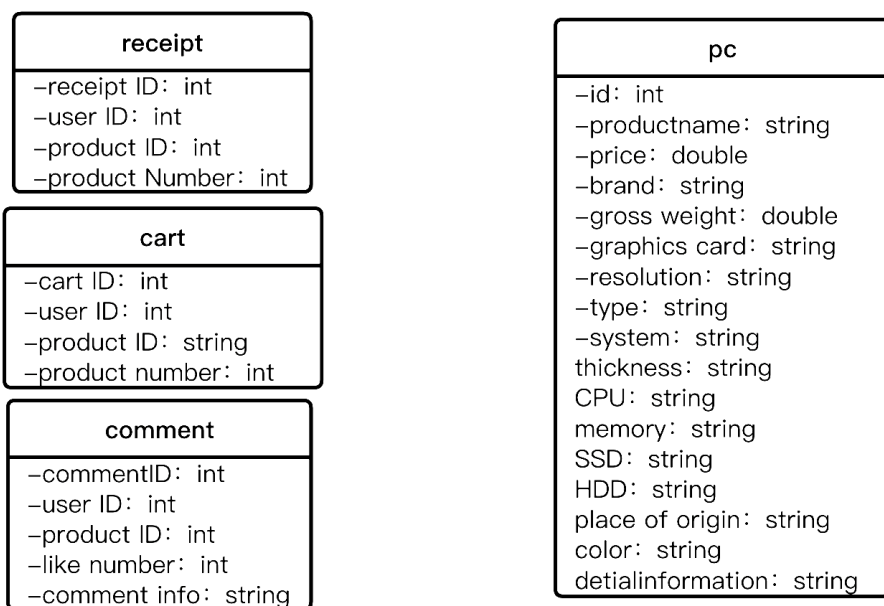
MVC 三层架构编写，上层 Controller 层负责连接外部接口与下层的 service 接口，中层 service 层负责给上层提供功能接口，提供 json 串形式的数据，调用数据库的相应功能实现相应的功能并封装成 json 形式的数据，Dao 层的接口给 service 层提供数据接口，Entity 层直接操作数据库实现对数据库的基本操作。

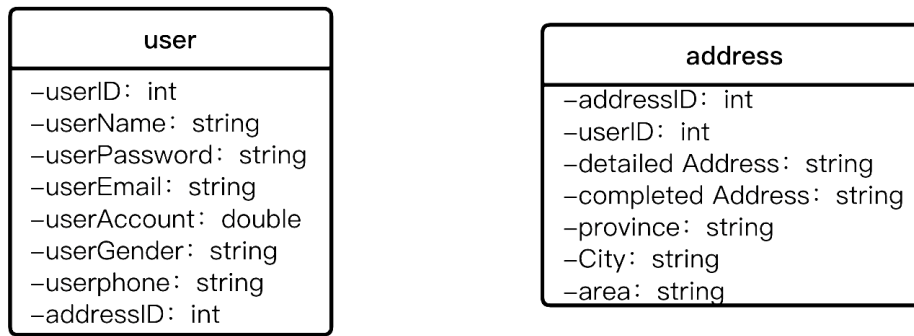
数据库在数据库设计，数据层 Entity 层给上层的 service 提供 8 个 dao 接口，每一个表对应一个 dao 接口，每一个接口对应一个 Entity 实体类负责直接实现对数据库执行相应的功能。

总体目录结构图：

数据库中的 8 个表对应 entity 中的 8 个实体类，可以实现基础的数据库的操作，查询某个属性值或是设置某个属性的值到数据库，这部分直接与数据库相连。dao 层的接口与 entity 相对应，负责给上层的 service 层提供相应的接口来实现相应的功能。service 负责实现各个页面的接口的功能，并将处理的数据封装成 json 串的形式返回给上层的 controller 层，controller 负责接收来自 web 前端的 url 的数据请求并调用 service 层的接口得到相关的 json 数据并返回给 web 前端。

entity 如下：





2 接口设计

2.1 用户/user

● 注册（post 方法）/register

参数:

参数名	类型	描述
email	String	邮箱信息，邮箱格式符合格式要求
password	String	用户密码

返回示例:

```
{
  "code": 200,
  "msg": "注册成功",
  "data": {
  }
}
```

● 登陆/login

参数

参数名	类型	描述
email	string	邮箱信息，邮箱格式符合格式要求
password	string	用户密码

返回示例:

```
{
  "code": 200,
  "msg": "登陆成功",
  "data": {
    "userId": "用户 ID"
    "userInfo": "用户信息"
    "nickName": "用户昵称"
  }
}
```

"Gender": "用户性别"

"image": "用户头像"

}

}

● 修改密码/reset-password

参数:

参数名	类型	描述
email	string	邮箱账号, 邮箱格式符合格式要求
password	string	用户密码

返回示例:

```
{
  "code": 200,
  "msg": "修改密码成功",
  "data": {
    "code": 200,
    "msg": "登陆成功",
    "data": {
      "userId": "用户 ID",
      "userInfo": "用户信息",
      "nickName": "用户昵称",
      "Gender": "用户性别",
      "image": "用户头像"
    }
  }
}
```

● 更改用户信息/update-user-info

参数

参数名	类型	描述
nickName	string	用户昵称
gender	string	用户性别
image	string	用户头像的网址

返回示例

```
{
  "code": 200,
  "msg": "更改用户信息成功",
  "data": {
    "code": 200,
    "msg": "登陆成功",
    "data": {
      "userId": "用户 ID",
      "userInfo": "用户信息",
      "nickName": "用户昵称"
    }
  }
}
```

```
"Gender": "用户性别"
"image": "用户头像"
}
}
}
```

● 给商品点赞/goods-like

参数

参数名	类型	描述
goodsId	int	商品信息
userId	int	用户信息

返回示例:

```
{
"code": 200,
"msg": "商品点赞成功",
"goodsId": "123456",
"likeNum": 10,
"likeState": true
}
```

● 增加地址/add_address

参数

参数名	类型	描述
userId	int	用户 id
name	string	用户收货名字，默认为昵称
phone	string	用户电话
detailedAddress	string	详细地址
completedAddress	string	完整地址
province	string	省
city	string	市
area	string	区

发送示例:

```
{
"name": "马力",
"phone": "17822091601",
"detailedAddress": "天津大学北洋园校区",
"completedAddress": "天津市天津大学北洋园校区",
"province": "天津市",
"city": "津南区",
"area": "海河教育园区"
}
```

返回示例:

```
{
"code": 200,
"msg": "增加成功",
}
```

```
"data":
{
}
}
```

● 获取地址/getaddress

参数

参数名	类型	描述
userid	int	需要获取用户的地址列表

返回示例

```
{
  "message":{
    "address":[
      {
        "name":"用户收货名称"
        "phone":"用户收获手机号"
        "detailedAddress":"用户的详细地址"
        "completedAddress":"用户的完整地址"
        "province":"省"
        "city":"市"
        "area":"区"
      },
      {

    },
    {

  },
  ]
  "meta":{
    "msg":"获取成功",
    "status":200
  }
}
```

● 修改地址/change_address

参数

参数名	类型	描述
userId	int	用户 id
name	string	用户收货名字，默认为昵称
phone	string	用户电话
detailedAddress	string	详细地址
completedAddress	string	完整地址
province	string	省

city	string	市
area	string	区

发送示例:

```
{
  "name": "马力",
  "phone": "17822091601",
  "detailedAddress": "天津大学北洋园校区",
  "completedAddress": "天津市天津大学北洋园校区",
  "province": "天津市",
  "city": "津南区",
  "area": "海河教育园区"
}
```

返回示例:

```
{
  "code": 200,
  "msg": "增加成功",
  "data": {
  }
}
```

● 删除地址/delete_address

参数

参数名	类型	描述
addressId	int	删除地址 id

返回示例:

```
{
  "code": 200,
  "msg": "删除地址成功",
  "data": {
  }
}
```

2.2 商品/goods

● 获取商品详细信息/goods-detail

参数:

参数名	类型	描述
goodsId	string	商品 id

返回示例:

```
{
  "message": {
```



```

"goodsid": "商品 id",
"goodsname": "商品名",
"goodsprice": "商品价格",
"goodscutprice": "商品打折价格",
"goodsnumber": "商品库存",
"goodsInfo": {
  "pics": [
    {
      "image_src": "商品图片地址数组"
    }
  ]
  "brand": "商品品牌",
  "grossweight": "商品净重",
  "graphics card": "显卡类型",
  "resolution": "分辨率",
  "type": "种类",
  "CPU": "",
  "detailinformation": "详细信息",
  "is_discut": "true/false",
},
},
"meta": {
  "msg": "获取成功",
  "status": 200
}
}

```

● 获取商品列表/goods-list

参数

参数名	类型	描述
priceRange	String	商品列表
currentPage	int	当前页码大小，从 0 开始
pagingSize	int	分页大小，默认值为 6
sortType	String	商品排序的类型

返回示例:

```

{
  "message": [
    {
      "goodsid": "商品 id",
      "goodsname": "商品名称",
      "goodsprice": "商品价格",
      "image_src": "商品图片地址，只有一张图片"
    }
  ],

```

```
{},
{}
],
"meta":{
"msg":"获取成功",
"status":200
}
}
```

2.3 评论/comment

● 写评论/write-comment

参数

参数名	类型	描述
goodsId	int	商品 ID
comment	string	评论的具体内容
userId	int	发表评论的用户的 ID
rate	int	星级评价有 0-5 星 5 个等级

发送示例

```
{
  "goodsId": "123456",
  "comment": "好好玩",
  "userId": "12456",
  "rate": 5
}
```

返回示例:

```
{
  "code": 200,
  "msg": "评价成功",
  "data":
  {
  }
}
```

● 给评论点赞/like-for-comment

参数

参数名	类型	描述
goodsId	int	商品 id
userId	int	用户 id
commentId	int	评论 id
likeState	string	点赞的状态

返回示例:

```
{
```

```

"goodsId": "100001",
"likeNum": 1,
"meta":
{
"msg": "点赞成功",
"status": 200
}
}

```

● 获取评论列表/comment-list

参数

参数名	类型	描述
goodsId	int	商品 id
pageSize	int	分页的大小非必需，默认是 6
curPage	int	当前页，默认是 0

返回示例:

```

{
"totalCommentNum": 1,
"curPage": 0,
"commentList": [
"commentId": " "
"list": [
{
"comment": "评论的内容"
"createAt": "评论的时间"
"likeNum": 1
}
],
"userInfo":
{
"image": " "
"gender":
"nickName": " " }
]
"meta":{
"msg": "获取成功",
"status": 200
}
}

```

2.4 购物车/cart

● 加入购物车/add_cart

参数

参数名	类型	描述
userId	int	用户 ID
goodsId	int	加入的商品 id
goodsNum	int	商品的数量

发送示例:

```
{
  "userId": " ",
  "goodsId": " ",
  "goodsNum": 1
}
```

返回示例:

```
{
  "code": 200,
  "msg": "加入购物车成功",
  "data":
  {
  }
}
{"listLength": 13, "curPage": 0, "cartList": [ "goodsId": "100004", "goodsNum": 1 ] }
```

● 删除购物车中商品/delete_cart

参数

参数名	类型	描述
userId	int	用户 id
goodsId	int	商品 id

发送示例:

```
{
  "userId": " ",
  "goodsId": " ",
}
```

返回示例:

```
{
  "code": 200,
  "msg": "删除购物车成功",
  "data":
  {
  }
}
```

● 获取购物车列表/get_cart

参数

参数名	类型	描述
userId	int	用户 id
pageSize	int	分页的大小, 默认为 6

curPage	int	当前页，默认为 0
---------	-----	-----------

返回示例:

```
{
  "listLength": 13,
  "curPage": 0,
  "cartList": [
    "goodsId": "100004",
    "goodsNum": 1
  ]
}
```

2.5 订单/order

● 新建订单/create-order

参数

参数名	类型	描述
addressId	int	收货地址 id
orderList	Object[]	订单列表

发送示例

```
{
  "userId": " ",
  "addressId": " ",
  "orderList": [
    {
      "goodsId": " ",
      "purchaseNum": 1
    }
  ],
}
```

返回示例:

```
{
  "code": 200,
  "msg": "创建订单成功",
  "data": {
  }
}
```

● 删除订单/delete_order

参数

参数名	类型	描述
orderId	int	订单编号

发送出 orderId 会返回操作结果，如果成功，返回成功，否则返回失败

● 支付订单/pay-for-order

参数

参数名	类型	描述
orderId	int	订单 id

支付订单时会返回当前账户余额

```
{ "balance": }
```

● 查找订单列表/order-list

参数

参数名	类型	描述
userId	int	需要查询用户的 id
pageSize	int	分页大小，默认为 6
currentPage	int	当前页，默认为 0

返回示例：

```
{
  "orderList": [{
    "orderId": " ",
    "totalAmount": 1785,
    "totalNum": 1,
    "goodsList": [{
      "_id": " ",
      "goodsId": " ",
      "goodsName": " ",
      "purchaseNum": 1,
      "unitPrice": ,
      "totalPrice": ,
      "discountPrice": ,
      "masterImg": " "
    }],
    "paymentState": "UNPAID"
  }]
}
```

2.6 账户/account

● 充值/recharge

参数：

参数名	类型	描述
rechargeKey	string	充值金额
verifyCode	string	验证码，本项目为一固定值

发送示例：

```
{ "rechargeKey": " ",
  "verifyCode": " " }
```

返回示例：

```
{ "balance": 12345 }
```

发出充值面额以及验证码，返回账户的余额。

● 获取余额/balance

参数:

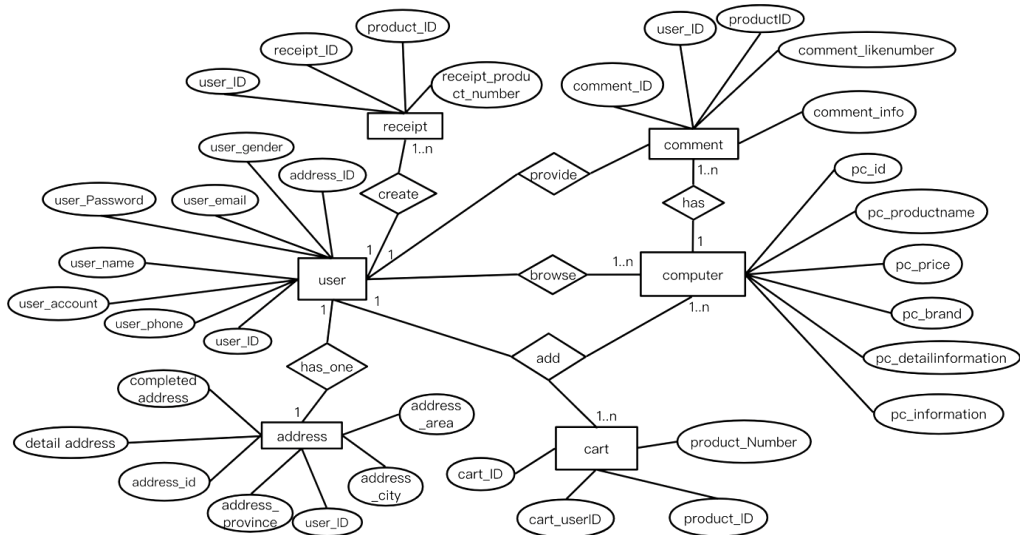
参数名	类型	描述
userId	int	用户的 id

发送用户的 id，获得该用户的账号的余额

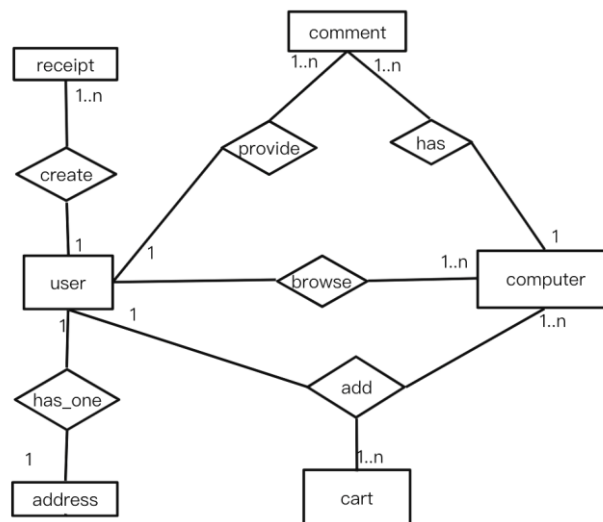
3 数据库设计

3.1E-R 图设计

1.完整 E-R 图设计



2.E-R 图结构（不带属性）



3.数据库关系说明

一个用户只能有一个地址，地址与用户之间是一对一的关系。用户可以创建多个订单，用户可以对购买过的商品进行评论，同样一个商品可以有条评论。用户可以浏览商品，并且将商品加入购物车。

3.2 数据表展示

1.user 表

列名	类型	属性
userid	int(11)	NOT NULL AUTO_INCREMENT PRIMARY KEY
userName	varchar(255)	NOT NULL
userPassword	varchar(255)	NOT NULL
user Email	varchar(255)	NOT NULL
user account	int(11)	NOT NULL
user gender	varchar(255)	DEFAULT NULL
user phone	varchar(255)	DEFAULT NULL
address ID	int(11)	NOT NULL

2.address 表

列名	类型	属性
address ID	int(11)	NOT NULL AUTO_INCREMENT PRIMARY KEY
user ID	int(11)	NOT NULL
detailed Address	varchar(255)	DEFAULT NULL
completed Address	varchar(255)	DEFAULT NULL
province	varchar(255)	DEFAULT NULL
City	varchar(255)	DEFAULT NULL
area	varchar(255)	DEFAULT NULL

3.cart 表

列名	类型	属性
cart ID	int(11)	NOT NULL AUTO_INCREMENT PRIMARY KEY
user ID	int(11)	NOT NULL
product ID	int(11)	NOT NULL
product number	int(11)	NOT NULL

4.comment 表

列名	类型	属性
commentID	int(11)	NOT NULL AUTO_INCREMENT PRIMARY KEY

user ID	int(11)	NOT NULL
product ID	int(11)	NOT NULL
like number	int(11)	NOT NULL
comment info	varchar(255)	DEFAULT NULL

5. receipt 表

列名	类型	属性
Receipt ID	int(11)	NOT NULL AUTO_INCREMENT PRIMARY KEY
user ID	int(11)	NOT NULL
product ID	int(11)	NOT NULL
product Number	int(11)	NOT NULL

6. pc 表

列名	类型	属性
Pc_id	int(11)	NOT NULL AUTO_INCREMENT
productname	varchar(255)	NOT NULL
price	double(10,2)	NOT NULL
brand	varchar(255)	NOT NULL
gross weight	double(10,2)	DEFAULT NULL
graphics card	varchar(255)	NOT NULL
resolution	varchar(255)	DEFAULT NULL
type	varchar(255)	DEFAULT NULL
system	varchar(255)	NOT NULL
thickness	varchar(255)	DEFAULT NULL
CPU	varchar(255)	NOT NULL
memory	varchar(255)	NOT NULL
SSD	varchar(255)	NOT NULL
HDD	varchar(255)	DEFAULT NULL
place of origin	varchar(255)	DEFAULT NULL
color	varchar(255)	DEFAULT NULL
picture	varchar(255)	DEFAULT NULL
detialinformation	varchar(255)	DEFAULT NULL