1. 系统工具及端口

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 端口 | 是否开通防火墙 | 可用账号 |
| SSH连接 | 22 | 是 | root/123456; |
| MariaDB | 3306 | 是 | root/123456; devsdus/123456; |
| Redis | 6379 | 是 | 指令：auth 123456（密码） |
| Nginx | 9500 | 是 |  |

1. 微服务名称及端口

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 服务代码 | 服务名称 | 服务发现名 | 端口 | 功能简述 |
| EurekaServer | 服务注册发现服务 |  | 9000 | 整个微服务框架的注册发现服务，用于发现框架中的其他微服务 |
| APIGateWay | 网关服务 |  | 9001 | 用于整个网站的网关路由，实现请求转发和负载均衡 |
| UserManageServer | 用户管理服务 | service-user/SERVICE-USER | 9100 | 1. 登录、注册； 2. 用户信息维护； 3. 用户收货地址管理； 4. ... |
| CommodityServer | 商品管理服务 | service-com/SERVICE-COM | 9101 | 1. 商品查询展示； 2. 商品上下架； 3. ... |
| OrderManageServer | 订单管理服务 | service-order/SERVICE-ORDER | 9102 | 1. 添加购物车； 2. 创建订单（直接购买）； 3. 修改购物车； 4. 删除购物车； 5. 创建订单（购物车结算）； 6. 提交订单； 7. ... |
|  |  |  |  |  |

1. APIGateWay网关路由规则

使用gateway网关，配置好之后，所有请求均走网关，URL头部为：[http://localhost:9001](http://192.168.40.132:9000/)

|  |  |  |
| --- | --- | --- |
| 微服务名 | 路由配置 | 服务发现名 |
| UserManageServer | /user/\*\*\* | SERVICE-USER |
| CommodityServer | /commodity/\*\*\* | SERVICE-COM |
| OrderManageServer | /order/\*\*\* | SERVICE-ORDER |
|  |  |  |

1. 各服务接口
2. UserManageServer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 暴露接口名 | 功能 | 请求方式 | 入参 | 返回说明 |
| login | 用户登录 | POST | {"tel":"手机号","password":"密码"} | 0 成功；-1 未注册； -2 信息有误 |
| register | 用户注册 | POST | {"tel":"手机号","password":"密码"} | 0 成功；-1 失败 |
| defaultaddress | 获取默认收货地址 | GET | userid | Json:{"address":"收货地址"} |
|  |  |  |  |  |
|  |  |  |  |  |

1. CommodityServer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 暴露接口名 | 功能 | 请求方式 | 入参 | 返回说明 |
| recommendindex | 首页商品推荐列表 | GET |  | Json:商品展示模型列表 |
| categoriesindex | 首页商品分类列表 | GET | categoryid | Json:商品展示模型列表 |
| recommendshop | 某店铺商品推荐列表 | GET | shpoid | Json:商品展示模型列表 |
| categoriesshop | 某店铺商品分类列表 | GET | Shpoid;categoryid | Json:商品展示模型列表 |
| cominfo | 某商品设定（上架）参数 | GET | shopid;categoryid;comid | Json:商品上架参数模型 |
| comdetials | 某商品的介绍详情 | GET | shopid;categoryid;comid | Json:{"address":"收货地址"} |
| comevaluation | 某店某商品的评价 | GET | shopid;categoryid;comid | Json:商品评价模型列表 |
|  |  |  |  |  |

1. OrderManageServer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 暴露接口名 | 功能 | 请求方式 | 入参 | 返回说明 |
| addtocart | 添加购物车 | POST | {  "userid": "用户账号",  "shpoid": "店铺编号",  "categoryid": "种类编号",  "comid": "商品编号",  "selparas": "选购的参数"  } | 0 成功；-1 失败 |
| createorderdirect | 创建订单，直接购买情况 | POST | {  "userid": "用户账号",  "shpoid": "店铺编号",  "categoryid": "种类编号",  "comid": "商品编号",  "selparas": "选购的参数"  } | 0 成功；-1 失败 |
| deletefromcart | 从购物车删除某商品，可选择多个 | POST | [{  "userid": "用户账号",  "shpoid": "店铺编号",  "cartid": "购物车信息编号"  },{}] | 0 成功；-1 失败 |
| changecart | 修改购物车信息，可多选 | POST | [{  "userid": "用户账号",  "shpoid": "店铺编号",  "cartid": "购物车信息编号",  "newnumber":"更改后数量"  }] | 0 成功；-1 失败 |
| createorderbycart | 创建订单，通过购物车 | POST | [{  "userid": "用户账号",  "shpoid": "店铺编号",  "cartid": "购物车信息编号"  }] | 0 成功；-1 失败 |
| commitorder | 提交订单 | POST | {  "userid": "用户账号",  "shpoid": "店铺编号",  "cartid": "购物车信息编号",  "comid":"商品编号",  "selparas":"选购的参数",  "address":"收货地址",  "remark":"备注"  } | 0 成功；-1 失败 |
|  |  |  |  |  |
|  |  |  |  |  |

1. 表设计

数据库名：myecom （均为小写）

后期表格采用分区方式进行管理，以店铺名进行分区。Mysql不支持自动增加表分区，所以只采用联合主键方式进行管理数据

1. 用户表
2. 用户个人信息表
3. 用户收货地址表
4. 店铺商品上架信息表
5. 商品参数表
6. 首页推荐商品表
7. 店铺推荐商品表
8. 商品详情表
9. 商品评价表
10. 购物车表
11. 订单表
12. 用户表 user

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| userid | Varchar(20) | × |  | 主键：不可修改，用户编号，用户注册时，会在此表中插入一条数据，初始userid和用户手机号相同；当某次用户修改手机号时，会修改tel字段，但是不会修改userid；当表中tel和userid均出现过某个手机号时，该手机号不能注册。 |
| tel | Varchar(20) | × |  | 注册时手机号，默认绑定手机号 |
| password | Varchar(20) | × |  | 密码 |
| ortherinfo | Varchar(50) | √ | 空字符 | 备用字段 |

1. 户个人信息表 userinfo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| userid | Varchar(20) | × |  | 主键：不可修改，用户编号 |
| tel | Varchar(20) | × |  | 注册时手机号，默认绑定手机号 |
| username | Varchar(20) | √ | userid | 用户昵称，初始为用户编号 |
| email | Varchar(50) | √ | 空字符 | 绑定邮箱 |
| ortherinfo | Varchar(50) | √ | 空字符 | 备用字段 |
|  |  |  |  |  |

1. 用户收货地址表 useraddress

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| userid | Varchar(20) | × |  | 用户编号，索引键，不可修改 |
| address | Varchar(100) | × |  | 收货地址 |
| ifdefault | int(1) | √ | 0 | 是否设为默认地址，0不是默认，1默认 |

1. 店铺商品上架信息表 commaininfo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| shopid | Varchar(20) | × |  | 所属店铺编号，分区键 |
| comid | Varchar(20) | × |  | 商品编号，聚集索引键 |
| categoryid | Varchar(20) | × |  | 商品种类编号 |
| showphoto | Varchar(100) | × |  | 展示用图片（访问地址） |
| shortmsg | Varchar(100) | × |  | 商品简介 |
| ifrecommend | Int(1) | √ | 0 | 是否推荐，0不推荐，1推荐 |
| addtime | datetime | × |  | 上架时间 |

1. 商品参数表 comtotalinfo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| comid | Varchar(20) | × |  | 商品编号，聚集索引键 |
| categoryid | Varchar(20) | × |  | 商品种类编号 |
| shopid | Varchar(20) | × |  | 所属店铺编号，分区键 |
| photos | Varchar(500) | × |  | 商品展示用图片（访问地址），默认至少一张 |
| comparas | Varchar(500) | × |  | 商品上架时设定的参数，以JSON格式保存，此字段可以考虑使用MongoDB保存 |

1. 首页推荐商品表（由平台维护） indexrecommend

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| comid | Varchar(20) | × |  | 商品编号，聚集索引键 |
| shopid | Varchar(20) | × |  | 所属店铺编号，分区键 |
| showphoto | Varchar(100) | × |  | 展示用图片（访问地址） |
| shortmsg | Varchar(100) | × |  | 商品简介 |
| begintime | Datetime | × |  | 展示开始时间 |
| endtime | Datetime | × |  | 展示结束时间 |

1. 店铺推荐商品表（由平台维护） indexrecommendofcategory

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| comid | Varchar(20) | × |  | 商品编号，聚集索引键 |
| categoryid | Varchar(20) | × |  | 商品种类编号 |
| shopid | Varchar(20) | × |  | 所属店铺编号，分区键 |
| showphoto | Varchar(100) | × |  | 展示用图片（访问地址） |
| shortmsg | Varchar(100) | × |  | 商品简介 |
| begintime | Datetime | × |  | 展示开始时间 |
| endtime | Datetime | × |  | 展示结束时间 |

1. 商品详情表 comdetials

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| comid | Varchar(20) | × |  | 商品编号，聚集索引键 |
| categoryid | Varchar(20) | × |  | 商品种类编号 |
| shopid | Varchar(20) | × |  | 所属店铺编号，分区键 |
| detials | Varchar(max) | √ | 空字符 | 商品详情，字符串表示，内容为xml格式的内容，前端直接解析 |

1. 商品评价表 evaluation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| orderid | Varchar(40) | × |  | 订单编号 |
| comid | Varchar(20) | × |  | 商品编号，聚集索引键 |
| categoryid | Varchar(20) | × |  | 商品种类编号 |
| shopid | Varchar(20) | × |  | 所属店铺编号，分区键 |
| goodlevel | Int(1) | √ | 5 | 好评分数，1-5,5最好 |
| evaluations | Varchar(500) | √ | 空字符 | 商品评价 |
|  |  |  |  |  |

1. 购物车表 shopcart

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| cartid | Varchar(40) | × |  | 购物车编号，主键 |
| userid | Varchar(20) | × |  | 不可修改，用户编号 |
| comid | Varchar(20) | × |  | 商品编号 |
| selectedparas | Varchar(500) | × |  | 选购参数，JSON格式参数，用于前端解析 |
| quantity | Int | × |  | 购买数量 |
| addtime | datetime | × |  | 加入购物车的日期 |

1. 订单表 userorder

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 字段类型 | 是否可空 | 默认值 | 说明 |
| orderid | Varchar(40) | × |  | 订单编号，主键 |
| userid | Varchar(20) | × |  | 不可修改，用户编号 |
| comid | Varchar(20) | × |  | 商品编号 |
| selectedparas | Varchar(500) | × |  | 选购参数，JSON格式参数，用于前端解析 |
| quantity | Int | × |  | 购买数量 |
| address | Varchar(100) | × |  | 收货地址 |
| recipient | Varchar(20) | × |  | 收货人姓名 |
| recipientphone | Varchar(20) | × |  | 收货人联系方式 |
| cost | Decimal(12,2) | × |  | 交易金额 |
| orderdate | datetime | × |  | 下单日期 |

1. 接口使用的model及json结构

1. 登录model

String tel;

String password;

登录json

{

"tel": "手机号",

"password": "密码"

}

2. 注册model

String tel;

String password;

注册json

{

"tel": "手机号",

"password": "密码"

}

3. 获取地址

String address;

获取地址 List<String> 的 json

4. 商品展示模型

String shopid;

String categoryid;

String comid;

String showphoto;//展示照片的url

String shortmsg;//商品简介

商品展示模型列表json

[

{

"shopid": "店铺编号",

"categoryid": "商品种类编号",

"comid": "商品编号",

"showphoto": "照片url",

"shortmsg": "商品简介"

}

]

5. 商品上架参数模型

String shopid;

String categoryid;

String comid;

List<String> photos;//商品的照片url列表

String comparas;//商品的上架参数，json格式

商品上架参数json

{

"shopid": "店铺编号",

"categoryid": "商品种类编号",

"comid": "商品编号",

"photolist": [

{

"showphoto": "照片url"

}

],

"comparas": "商品的上架参数，json数据的String"

}

6. 商品评价模型列表

String shopid;

String categoryid;

String comid;

List<EvaluationModel> evaluations;

商品评价模型列表json

{

"shopid": "店铺编号",

"categoryid": "商品种类编号",

"comid": "商品编号",

"evaluations": [

{

"userid": "用户编号",

"orderid": "订单编号",

"goodlevel": 5,

"evaluation": "评价"

}

]

}

7. 用户选购模型

String userid;

String shopid;

String categoryid;

String comid;

String selparas;

用户选购模型json

{

"userid": "用户账号",

"shpoid": "店铺编号",

"categoryid": "种类编号",

"comid": "商品编号",

"selparas": "选购的参数，json数据的String"

}

8. 删除购物车模型 购物车结算模型 通用

String userid;

List<String> cartids;

删除购物车 购物车结算 json

{

"userid": "用户账号",

"cartids": [

{

"cartid": "购物车编号"

}

]

}

9. 修改购物车模型

String userid;

List<CartModel> carts;

修改购物车json

{

"userid": "用户账号",

"carts": [

{

"cartid": "购物车编号",

"newnumber": "更改后数量"

}

]

}

10. 下单模型

String userid;

String shopid;

String cartid;

String comid;

String selparas;

String address;

String remark;

BigDecimal cost;

下单json

{

"userid": "用户账号",

"shpoid": "店铺编号",

"cartid": "购物车信息编号",

"comid": "商品编号",

"selparas": "选购的参数",

"address": "收货地址",

"remark": "备注",

"cost": 99.99

}