秒杀系统概要设计

|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [√] 正式发布  [ ] 正在修改 | 文件标识： |  |
| 当前版本： | 1.0 |
| 作 者： | 周智祥 |
| 完成日期： |  |

文档修订记录

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 版本编号或者更改记录编号 | \*变化  状态 | 简要说明（变更内容和变更范围） | 日期 | 变更人 | 审核日期 | 审核人 |
| V1.0 | A | 初始创建 |  | 周智祥 |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

\*变化状态：A——增加，M——修改，D——删除

文档审批信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 审批人 | 角色 | 审批日期 | 签字 | 备注 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[1、引言 5](#_Toc496110274)

[1.1 目的 5](#_Toc496110275)

[1.2 范围 5](#_Toc496110276)

[1.3 术语定义 5](#_Toc496110277)

[1.4 相关文档 6](#_Toc496110278)

[1.5 参考资料 6](#_Toc496110279)

[2、 总体设计 7](#_Toc496110280)

[2.1 需求规定 7](#_Toc496110281)

[2.2总体结构设计 7](#_Toc496110282)

[2.1.1层次结构图 7](#_Toc496110283)

[2.1.2总体架构图 7](#_Toc496110284)

[2.1.3技术架构 7](#_Toc496110285)

[2.1.4功能需求与程序 7](#_Toc496110286)

[2.1.5人工处理过程 7](#_Toc496110287)

[2.1.6未解决问题 8](#_Toc496110288)

[2.3运行环境设计 8](#_Toc496110289)

[2.4子系统清单 9](#_Toc496110290)

[2.5功能模块清单 9](#_Toc496110291)

[2.6概念和处理流程 10](#_Toc496110292)

[2.6.1系统概念 10](#_Toc496110293)

[2.6.2处理流程 10](#_Toc496110294)

[3、模块功能分配 11](#_Toc496110295)

[3.1专用模块功能分配 11](#_Toc496110296)

[3.2公用模块功能分配 11](#_Toc496110297)

[3.3模块间关系 12](#_Toc496110298)

[4、数据结构 12](#_Toc496110299)

[4.1数据库表名清单 12](#_Toc496110300)

[4.2数据库表之间关系说明 13](#_Toc496110301)

[4.3数据库表的详细清单 13](#_Toc496110302)

[4.4视图的设计 13](#_Toc496110303)

[4.5数据结构和程序的关系 13](#_Toc496110304)

[4.6主要算法设计 14](#_Toc496110305)

[4.7其它数据结构设计 14](#_Toc496110306)

[5、接口设计 14](#_Toc496110307)

[5.1外部接口 14](#_Toc496110308)

[5.1.1XX接口设计 14](#_Toc496110309)

[5.1.2XX接口设计 15](#_Toc496110310)

[5.2用户接口 15](#_Toc496110311)

[5.3内部接口 15](#_Toc496110312)

[6、运行设计 15](#_Toc496110313)

[6.1运行模块组合 15](#_Toc496110314)

[6.2运行控制 15](#_Toc496110315)

[6.3运行时间 16](#_Toc496110316)

[7、 出错处理 16](#_Toc496110317)

[7.1出错信息和补救措施 16](#_Toc496110318)

[7.2系统维护设计 16](#_Toc496110319)

[8、 其它设计 16](#_Toc496110320)

[9、附录 17](#_Toc496110321)

[9.1附录A－相关过程 17](#_Toc496110322)

[9.2附录B－相关规程 17](#_Toc496110323)

[9.3附录C－相关指南 17](#_Toc496110324)

[9.4附录D－相关模板列表 17](#_Toc496110325)

1、引言

## 目的

本文档的主要任务是将需求分析文档转换为软件结构和数据结构。设计软件结构的具体任务是：将一个复杂系统按功能进行模块划分、建立模块的层次结构及调用关系、确定模块间的接口及人机界面等。数据结构设计包括数据特征的描述、确定数据的结构特性、以及数据库的设计。

本说明书的预期读者为：系统设计者、系统开发员、测试人员。

业务需求目的说明：

## 范围

本文档的上游（依据的基线）是需求分析规格书，它的下游是系统详细设计说明书，并为详细设计说明书提供测试的依据。

软件概要设计的范围是：软件系统总体结构、核心流程处理、外部接口、内部接口、主要部件功能分配、全局数据结构以及部件之间的接口等方面的内容。

## 术语定义

|  |  |  |
| --- | --- | --- |
| **序号** | **术语名称** | **术语定义** |
| 1 | 总体结构 | 软件系统的总体逻辑结构。按照不同的设计方法，有不同的总体逻辑结构。若采用传统的面向功能或面向数据的结构化设计方法，则总体逻辑结构为一树形的功能模块结构图。若采用面向对象或面向部件（组件）的设计方法，则总体逻辑结构为部件（组件）的组装图。 |
| 2 | 外部接口 | 本软件系统与其它软件系统之间的接口，接口设施可以是中间件。接口描述包括：传输方式、带宽、数据结构、传输频率、传输量（兆/秒）、传输协议。 |
| 3 | 数据结构 | 数据结构包括：关系数据库表的结构、对象数据库表的结构、变量说明。 |
| 4 | 概念数据模型 | 关系数据库的逻辑设计模型，叫概念数据模型。主要内容包括一张逻辑E--R图及其相应的数据字典。 |
| 5 | 物理数据模型 | 关系数据库的物理设计模型，叫物理数据模型。主要内容包括一张物理表关系图及其相应的数据字典。 |
| 6 | 视图 | 在基表之上建立的一张虚表，叫视图，它具有物理表的许多性质，在授权上很有用。 |
| 7 | 角色 | 数据库中享有某些特权操作的用户，叫角色。 |
| 8 | 子系统 | 具有相对独立功能的小系统叫子系统。一个大的软件系统可以划分为多个子系统，每个子系统可由多个模块或多个部件组成。 |
| 9 | 模块 | 具有功能独立、能被调用的信息单元叫模块。模块是结构化设计中的概念。 |
| 10 | 部件（组件） | 具有功能独立、能被调用的、且已包装的信息单元叫部件（组件）部件是面向对象设计中的概念。 |
| 11 | 内部接口 | 软件系统内部各子系统之间、各部件之间、各模板之间的接口，叫内部接口。接口描述包括：调用方式、入口信息、出口信息等。 |
| 12 | 相关文件 | 相关文件是指：当本文件内容变更后，可能引起变更的其它文件。如需求分析报告、详细设计说明书、测试计划、用户手册。 |
| 13 | 参考资料 | 参考资料是指：本文件书写时用到的其它资料。如各种有关规范、模板、标准、准则。 |
| 14 | 受理终端 | 参与移动支付交易的受理机具，包括扫码业务终端、金融POS终端等 |
| 15 | 秒杀 | 同一时间大量用户抢购同一个商品 |

## 相关文档

|  |  |  |  |
| --- | --- | --- | --- |
| 文档名称 | 作者 | 文件编号、版本 | 资料存放地点 |
| 系统详细设计说明书 |  |  |  |
| 源程序清单 |  |  |  |
| 测试计划及报告 |  |  |  |
| 用户使用手册 |  |  |  |

## 参考资料

|  |  |  |  |
| --- | --- | --- | --- |
| 资料名称 | 作者 | 文件编号、版本 | 资料存放地点 |
|  |  |  |  |
|  |  |  |  |

# 总体设计

## 需求规定

* 不能少买
* 不能超卖
* 系统正常运行，不能出行卡顿或者down机现象

## 2.2 架构设计

1. nginx为前端负载均衡层
2. gateway为一层网关层，负责限流、权限校验等处理
3. 微服务业务层，负责业务逻辑处理，比如扣减商品，下单等
4. nacos作为注册中心和配置中心

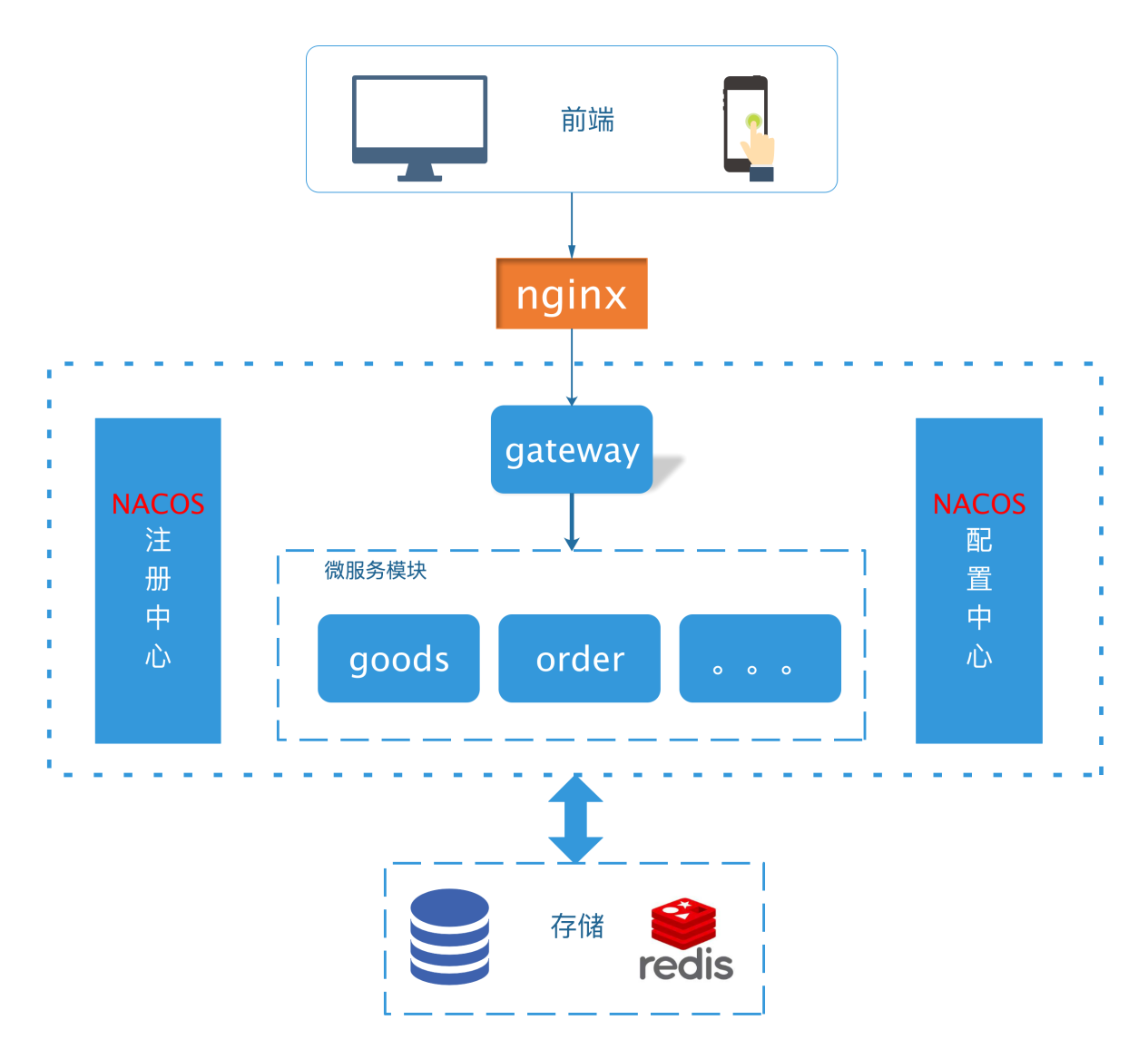


图1总体架构图

## 2.3 运行环境设计

该软件系统的运行环境：

1. 硬件平台：

服务器的最低配置要求：16核2.4GHz的CPU、32G内存

工作站的最低配置要求：系统Centos7.2、4核2.4GHz的CPU、4G内存

1. 软件平台：

服务器操作系统 ：Centos7.2

数据库管理系统：Mysql7.X

中间件

|  |  |  |
| --- | --- | --- |
| 名称 | 描述 | 版本 |
| redis | 缓存DB | V4.0 |
| nacos | 配置中心和注册中心 | V1.2 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

客户端的操作系统：XP/Win7/Win8/Win10/Mac等

客户端的平台软件：Android/iphone/IE/Chrome/360浏览器等

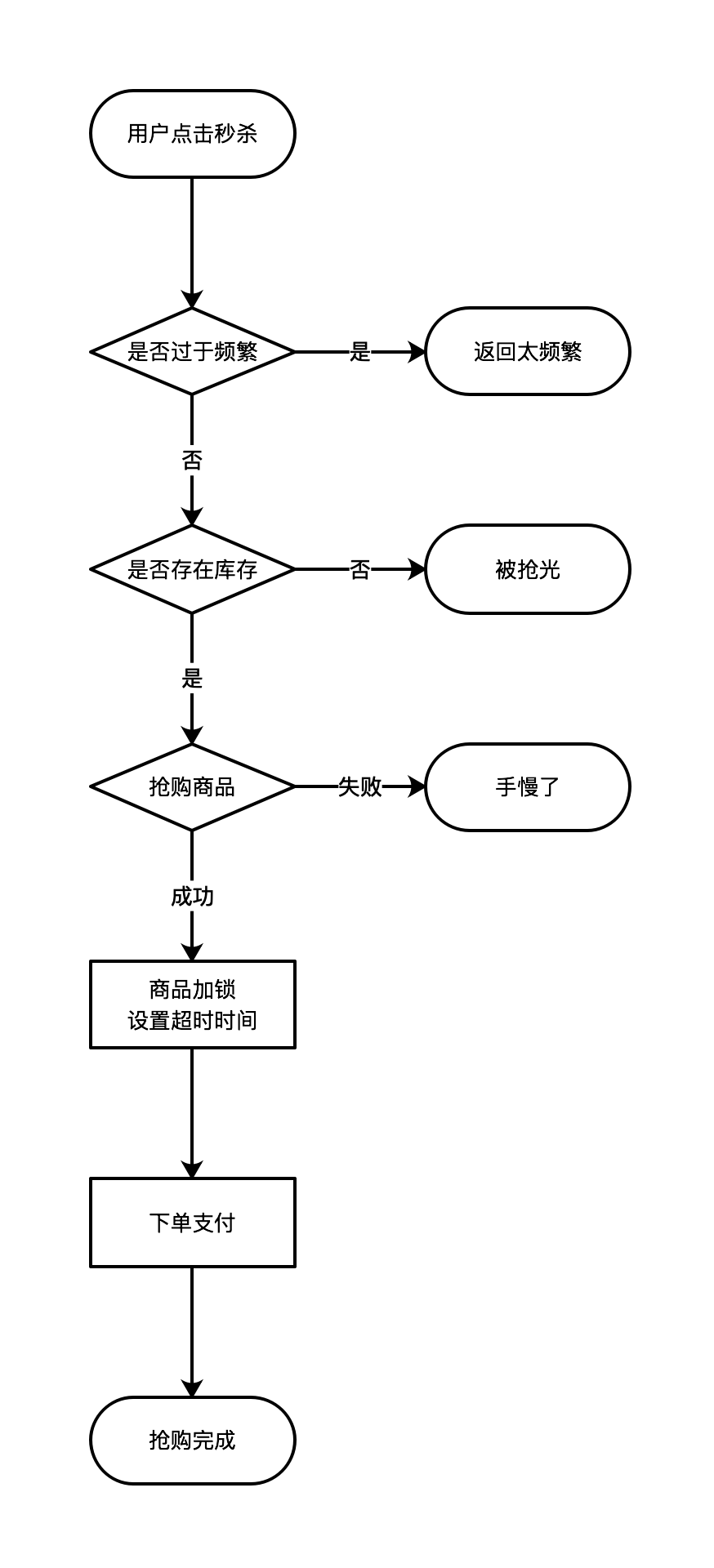
1. 网络平台

通信协议：HTTP、TCP/IP、FTP、WebService等

通信带宽：200Mps

## 2.4 业务逻辑

### 2.4.1用户抢购



### 2.4.2超卖处理

超卖问题解决网上一般有几种处理的方式：悲观锁，分布式锁，乐观锁，队列串行化，Redis原子操作

根据项目的情况可以选择最优的情况。本秒杀方案使用redis缓存+乐观锁的方式。其中redis缓存是保证查询判断等步骤不到db。

### 2.4.3超时处理

超时未支付订单，使用redis的zset结构+分布式定时任务，获取超时订单，并完成库存回滚，和修改订单状态

# 3、模块功能分配

具有功能独立、能被调用的信息单元叫模块。模块是结构化设计中的概念，部件是面向对象设计的概念。

模块功能分配的目的，就是为了将具有相同功能的模块合并，从中提取公用模块，形成公用部件，作为本系统的公用资源，甚至作为公司级组织的公用资源，从而优化系统设计，加快开发速度，提高开发质量。

## 3.1模块划分

|  |  |  |  |
| --- | --- | --- | --- |
| **模块编号** | **模块名称** | **模块详细功能分配** | **模块的接口标准** |
| Micro-gateway | 网关 |  |  |
| Micro-goods | 商品 |  | restful |
| Micro-order | 订单 |  | restful |
| Micro-common | 公共模块 |  |  |

## 3.2模块间关系

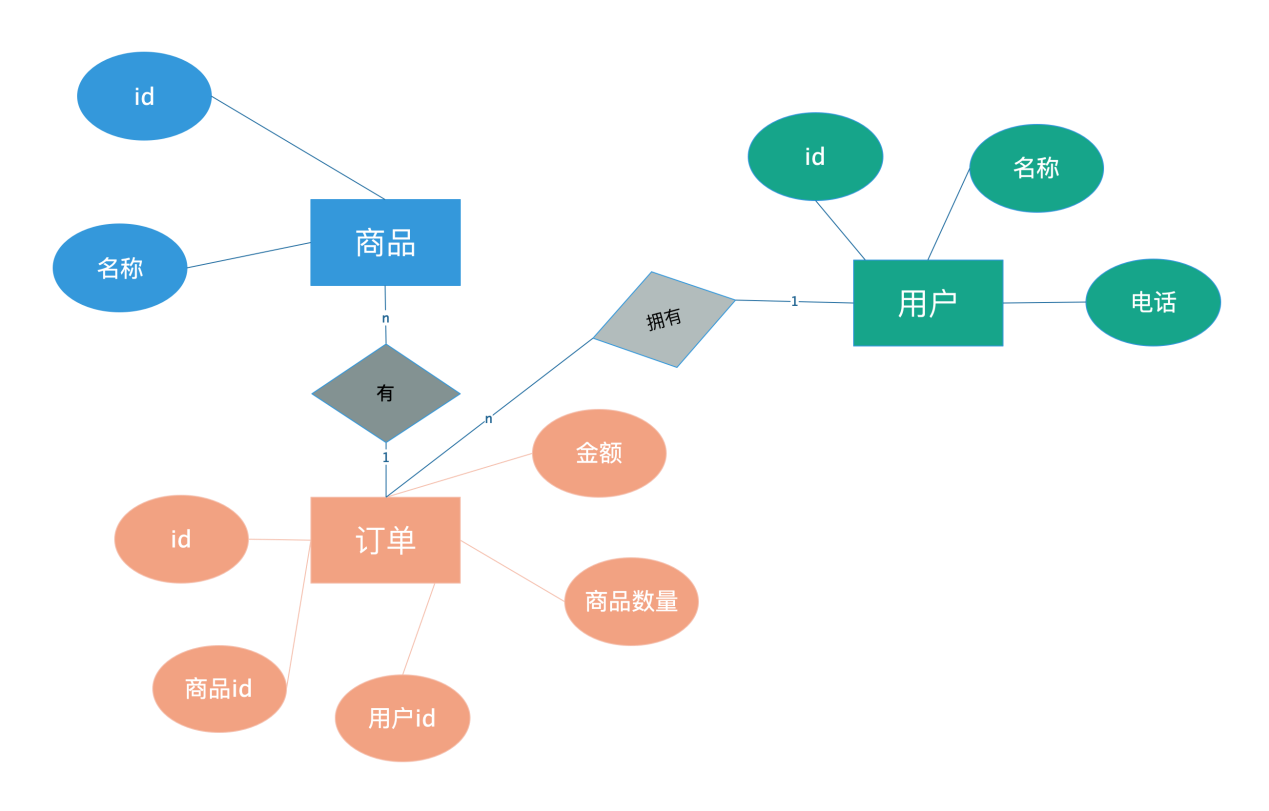
# 4、数据结构

数据库设计的基本原则是“三少”：一个数据库中表的个数越少越好，一个数据库中主键的个数越少越好，一个表中的字段个数越少越好。数据库设计的实用原则是：在数据冗余和处理速度之间找到合适的平衡点。

## 4.1数据库表名清单

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **中文表名** | **英文表名** | **表功能说明** |
| 1 | 商品表 | t\_goods |  |
| 2 | 订单表 | t\_order |  |
| 3 | 用户表 | t\_user |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |
| 11 |  |  |  |
| 12 |  |  |  |
| 13 |  |  |  |
| 14 |  |  |  |
| 15 |  |  |  |
| 16 |  |  |  |
| 17 |  |  |  |
| 18 |  |  |  |
| 19 |  |  |  |
| 20 |  |  |  |
| 21 |  |  |  |
| 22 |  |  |  |
| 23 |  |  |  |
| 24 |  |  |  |
| 25 |  |  |  |
| 26 |  |  |  |
| 27 |  |  |  |

## 4.2数据库表之间关系说明



## 4.3数据库表的详细清单

参考数据库设计文档6.2章节

## 4.4视图的设计

参考数据库设计文档6.3章节

## 4.5数据结构和程序的关系

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **中文表名** | **英文表名** | **程序模块关系描述** |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |
| 11 |  |  |  |
| 12 |  |  |  |
| 13 |  |  |  |
| 14 |  |  |  |
| 15 |  |  |  |
| 16 |  |  |  |
| 17 |  |  |  |
| 18 |  |  |  |
| 19 |  |  |  |
| 20 |  |  |  |
| 21 |  |  |  |
| 22 |  |  |  |
| 23 |  |  |  |
| 24 |  |  |  |
| 25 |  |  |  |
| 26 |  |  |  |
| 27 |  |  |  |

## 4.6主要算法设计

## 4.7其它数据结构设计

# 接口设计

## 5.1商品详情接口

接口名称：商品详情

接口内容：更加商品id获取商品详细信息

接口的数据结构：

接口协议：restful

## 5.2抢购接口

接口名称：秒杀抢购商品

接口内容：用户抢购指定商品

接口的数据结构：

接口协议：

## 5.3支付回调接口

接口名称：支付回调

接口内容：支付完成，接受支付平台支付结果通知回调

接口的数据结构：

接口协议：

# 运行设计

生产部署，可以使用docker+kubernetes，其优点可以快速新增节点，也可以在其框架上比较轻松完成灰度发布等。

## 6.1运行模块组合

|  |  |  |
| --- | --- | --- |
| 模块名称 | 运行环境和条件 | 依赖条件 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## 6.2运行控制

|  |  |  |
| --- | --- | --- |
| 运行名称 | 控制方法 | 操作步骤 |
|  |  |  |

## 6.3运行时间

|  |  |  |
| --- | --- | --- |
| 运行名称 | 所占资源 | 时间 |
|  |  |  |

# 出错处理

## 7.1出错信息和补救措施

列出每种可能出现的出错或故障出现时，系统输出信息的形式、含义。

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 名称 | 输出信息 | 处理策略 |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |

## 7.2系统维护设计

# 其它设计

# 9、附录

## 9.1附录A－相关过程

## 9.2附录B－相关规程

## 9.3附录C－相关指南

## 9.4附录D－相关模板列表