# 1、引言

## 1.1 编写目的

编写概要设计文档，是为了指导后续的详细设计

## 1.2 面向对象

架构师、高级工程师、初中级工程师

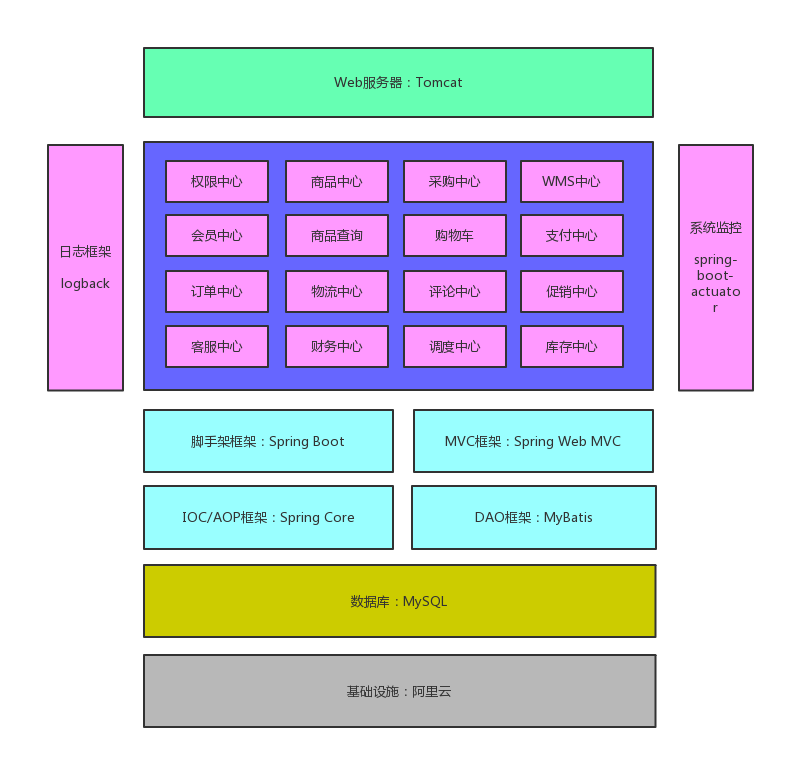
## 1.3 参考文档

1、《电商系统v1.0版本产品需求文档》

2、《电商系统v1.0版本需求分析文档》

# 2、逻辑架构

## 2.1 逻辑架构图



## 2.2 设计细节

### 2.2.1 Web服务器

采用tomcat作为web服务器，不引入apache或者是nginx这种服务器，因为第一个版本，预计的用户数量也就是100个内测用户，并发量基本低于1/s。系统主要是为了跑通咱们的系统逻辑，进行内测。所以说为了开发和部署的方便以及效率起见，直接采用tomcat作为web服务器。

### 2.2.2 业务系统

直接将16个中心（权限中心、商品中心、采购中心、WMS中心、会员中心、商品查询、购物车、支付中心、订单中心、物流中心、评论中心、促销中心、客服中心、财务中心、调度中心、库存中心）全部放在一个单块应用中。

不做任何的系统拆分，全部采取B/S架构，接口对外提供服务。因为刚开始我们开发人员很少，不需要做太多的服务拆分，全部耦合在一个服务中，开发所有的代码，效率反而是最高的。

### 2.2.3 脚手架框架

采用目前业界最新最热门最流行的spring boot技术，作为脚手架框架，利用spring boot与其他的框架技术快速的整合。同时spring boot的可扩展性非常好，未来如果要用到缓存、mq等其他的技术，都可以快速的跟spring boot脚手架进行整合。

而且，团队已经初步对spring boot完成了技术调研，spring boot 20%的核心功能都已经掌握的很扎实了。

### 2.2.4 MVC框架

采取业界最流行的spring web mvc框架即可

### 2.2.5 IOC/AOP框架

采取业界最流行的spring core框架即可

### 2.2.6 ORM框架

采取业界最流行的mybatis3框架，不采用纯粹的ORM框架，hibernate，是因为类似hinernate那样的框架太重了，黑盒，自动生成的那个SQL性能较差。所以不建议采用hibernate。反而是mybatis框架基本可以自己手写sql语句，灵活，通用，很方便。

### 2.2.7 数据库

采用MySQL作为数据库，因为oracle数据库到了后期使用要付费，费用高昂，而且扩展性不是很好。目前互联网行业最通用的都是采用mysql数据库，哪怕是后面数据量大了，要分库分表，也有很多成熟的方案可以用。

### 2.2.8 基础设施

机器，网络，基础设施，采用阿里云，因为目前行业里国内，基本上就是这个阿里云的产品最好，稳定性，技术支持能力最好。我们作为一个初创型团队，不应该直接自己花很多的钱去采购和运维管理自己的服务器，那样的话成本太高了，而且很麻烦。

### 2.2.9 日志框架

采用spring boot默认支持的logback作为日志框架即可

### 2.2.10 系统监控

第一个版本主要是内测，不需要复杂的系统监控，只需要使用spring boot actuator提供的简单的健康检查监控即可。

# 3、运行架构

## 3.1 系统核心流程

### 3.1.1 采购流程时序图

[采购流程时序图](https://www.processon.com/diagraming/5a77cbb8e4b059c41ab4ea8a)

### 3.1.2 购物流程时序图

[购物流程时序图](https://www.processon.com/diagraming/5a77d33ce4b059c41ab50763)

### 3.1.3 退货流程时序图

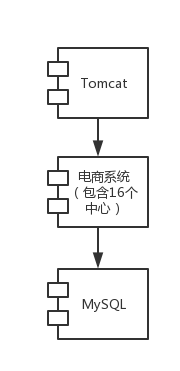
[退货流程时序图](https://www.processon.com/diagraming/5a77d821e4b0812a0f065bc0)

## 3.2 功能时序图

功能的运行流程与需求文档中几乎保持一致，所以在这里就不画了

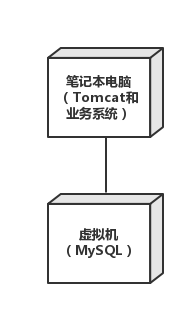
# 4、物理架构

## 4.1 软件架构

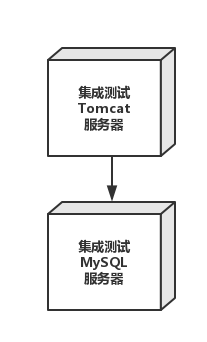


## 4.2 硬件架构

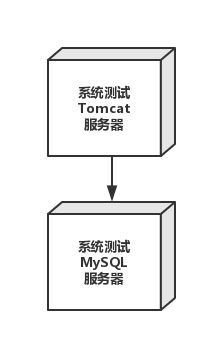
### 4.2.1 开发环境



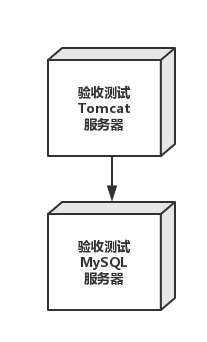
### 4.2.2 集成测试环境



### 4.2.3 系统测试环境



### 4.2.4 验收测试环境



### 4.2.5 线上环境

