

Design and Implementation of Haitian Online Shopping Mall Based on the Integration of Three Frameworks of SSM

Liangfeng Zhang, Yimin Lin

Cunjin College, Guangdong Ocean University, Zhanjiang Guangdong
Email: forrily@126.com

Received: Dec. 6th, 2017; accepted: Dec. 22nd, 2017; published: Dec. 29th, 2017

Abstract

Along with the gradual deepening of regional economic integration, regional and inter-regional economic exchanges more closely, the development of the logistics industry directly leads to a series of transactions between different regions, and makes the Zhanjiang Haitian logistics industrial park construction much wiser and more intellectualized. We should optimize the industrial development environment of the logistics industry park and introduce the modern information technology to realize the agglomeration effect of resources and functions. It will make the transformation and upgrades of the Haitian logistics industry park come true. And then, we can let it catch the "Internet plus" express, which will make Zhanjiang Haitian logistics industrial park more intellectualized and intelligent. This paper uses the idea of object-oriented programming, through the UML modeling with the function of mind map, through the integration of SSM three frameworks, before and after the end of separation development, the use of Spring, SpringMVC, MyBatis, Dubbo, Redis, JSON SolrCloud, technology, completed the design and implementation of the Zhanjiang Haitian.

Keywords

Spring, SpringMVC, MyBatis, Online Shopping Mall

基于SSM三大框架整合的海田在线商城的设计与实现

张良峰, 林益民

广东海洋大学寸金学院, 广东 湛江
Email: forrily@126.com

收稿日期: 2017年12月6日; 录用日期: 2017年12月22日; 发布日期: 2017年12月29日

摘要

伴随着区域经济一体化格局的逐步深化, 区域与区域之间的经济交流更为密切, 物流产业的发展直接催生不同地区之间一系列的商品交易, 而湛江海田物流产业园区建设尚且处于一个逐步发展的传统阶段。我们应优化物流产业园的产业发展环境, 引入现代信息技术实现资源与功能的集聚效应, 实现海田物流产业园的转型升级, 使之搭上“互联网+”的快车, 让湛江海田物流产业园更加“智慧化”和“智能化”。本文研究使用面向对象编程的思想, 通过UML建模结合功能思维导图, 通过整合SSM三大框架, 前后端分离开发, 使用Spring、SpringMVC、MyBatis、Dubbo、SolrCloud、Redis、JSON等开发技术, 完成了湛江海田在线商城的设计与实现。

关键词

Spring, SpringMVC, MyBatis, 在线商城

Copyright © 2017 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 绪论

1.1. 课题的研究背景

湛江海田物流产业园是湛江市深入贯彻对赤坎的产业定位和实施“三产兴区”发展战略的创新举措, 也是赤坎区发展第三产业和经济发展扩容提质的主要载体。

湛江海田物流产业园位于赤坎区东北部的海田片区, 紧靠城市主干道, 海、陆、空交通运输齐全、便利。目前, 产业园内的商贸物流已具有相当规模, 成为粤西最大的商贸物流聚集区, 形成了建材、汽车、水产三大支柱产业等 18 个专业市场, 拥有 3700 家各类大型商户, 2015 整个湛江海田物流产业园的年产值达 110 亿元人民币。经过充分的市场调研及与湛江市政府和海田物流产业园管委会的座谈, 我们发现在当前“互联网+”概念的冲击下, 湛江海田物流产业园的各类经营仍处于非常传统的商业运营, 整个产业园综合管理效能不高, 亟待产业的转型升级, 尤其是要引入现代的互联网思维, 实现线下向线上转型升级。

1.2. 研究目的及意义

伴随着区域经济一体化格局的逐步深化, 区域与区域之间的经济交流更为密切, 物流产业的发展直接催生不同地区之间一系列的商品交易, 而我国当前很多的物流产业园区建设尚且处于一个逐步发展的传统阶段。物流业的布局及物流成本的高低, 很大程度上制约着现代商贸企业的发展。因此, 我们应优化物流产业园的产业发展环境, 引入现代信息技术实现资源与功能的集聚效应, 实现传统物流业的转型升级, 使之搭上“互联网+”的快车, 让物流发展更加“智慧化”和“智能化” [1]。

本文的研究内容, 使用面向对象编程的思想, 通过 UML 建模[2]结合功能思维导图, 基于 SSM 三大框架的整合思路[3]实现海田在线商城的设计与实现。

2. 需求分析

软件需求分析是系统必须完成的事, 以及必须具备的品质, 具体来说, 需求分析包括功能性需求、非功能性需求和设计约束三个方面。软件需求分析的准确性对于海田在线商城实现的质量有着决定性的作用, 也是开发团队的指南针。本文的需求分析采用的是统一建模语言(UML)中的用例图来展示用户的需求, 描述用户和系统如何协作的功能和非功能性需求。用思维导图来描述海田在线商城核心子系统的功能需求。

2.1. 用户需求

用户的难题就是我们的课题。海田在线商城的实现旨在创建一个海田物流产业园区线上服务闭环, 形成一个安全可靠、方便易用的多点互通的电子商务交易平台。实现大众客户、企业客户、会员可以在商城上搜索商品, 浏览商品, 交易商品等; 供应商、厂商、零售商、品牌直营店可以在上平台发布商品、管理、售卖或者交易商品等; 湛江海田物流产业园管委会、系统管理员等都能在平台上操作、管理的一个线上产业园平台, 同时保证系统正常运行。

2.1.1. 用户需求分析

在项目的初始阶段, 举办了项目成员对于海田在线商城用户需求的讨论会, 根据目前整个互联网的发展趋向, 初步确定了以下几个方面。

- 1、功能齐全的商品展现平台和仓库管理系统;
- 2、灵活的订单处理机制;
- 3、支持多选的线上支付的功能及可选快递服务(特殊商品同城配送服务);
- 4、提供活动窗口, 广告投放窗口, vr 店铺展窗口;
- 5、基于平台各类数据的分析为普通用户提供商铺推荐;
- 6、完善的管理功能, 按照不同性质的用户进行权限分配;
- 7、开放的平台投资窗口。

2.1.2. 用例图

在需求分析初始阶段, 为了明确项目的目标及参与者, 更好的与项目干系人进行沟通, 我们以用例图的形式来描述用例和参与者及其之间的关系。用例图提供了可视的语境图, 明确的阐述外部参与者及其对系统的使用; 同时, 展示了系统的边界、位于边界之外的事物以及系统如何别使用; 作为沟通的工具, 概括了系统的全貌、系统以及参与者的行为。系统用例图如图 1 和图 2 所示。

用例定义了参与者与海田在线商城平台之间的关系, 用户需求由系统来完成实现, 前端系统则主要针对大众用户和会员用户, 后台系统则对系统管理员和商城后台操作员开放。

2.2. 功能需求

2.2.1. 功能架构图

如图 3 所示, 海田在线商城主要由前台 12 个基本功能和后台 12 个主要功能系统组成, 完成电商平台的基本用户需求。

2.2.2. 主要功能开发思维导图(图 4~图 8)

思维导图是一种将思维形象化的方法。思维导图在海田在线商城的开发中能引导整个项目的开发不偏离方向。

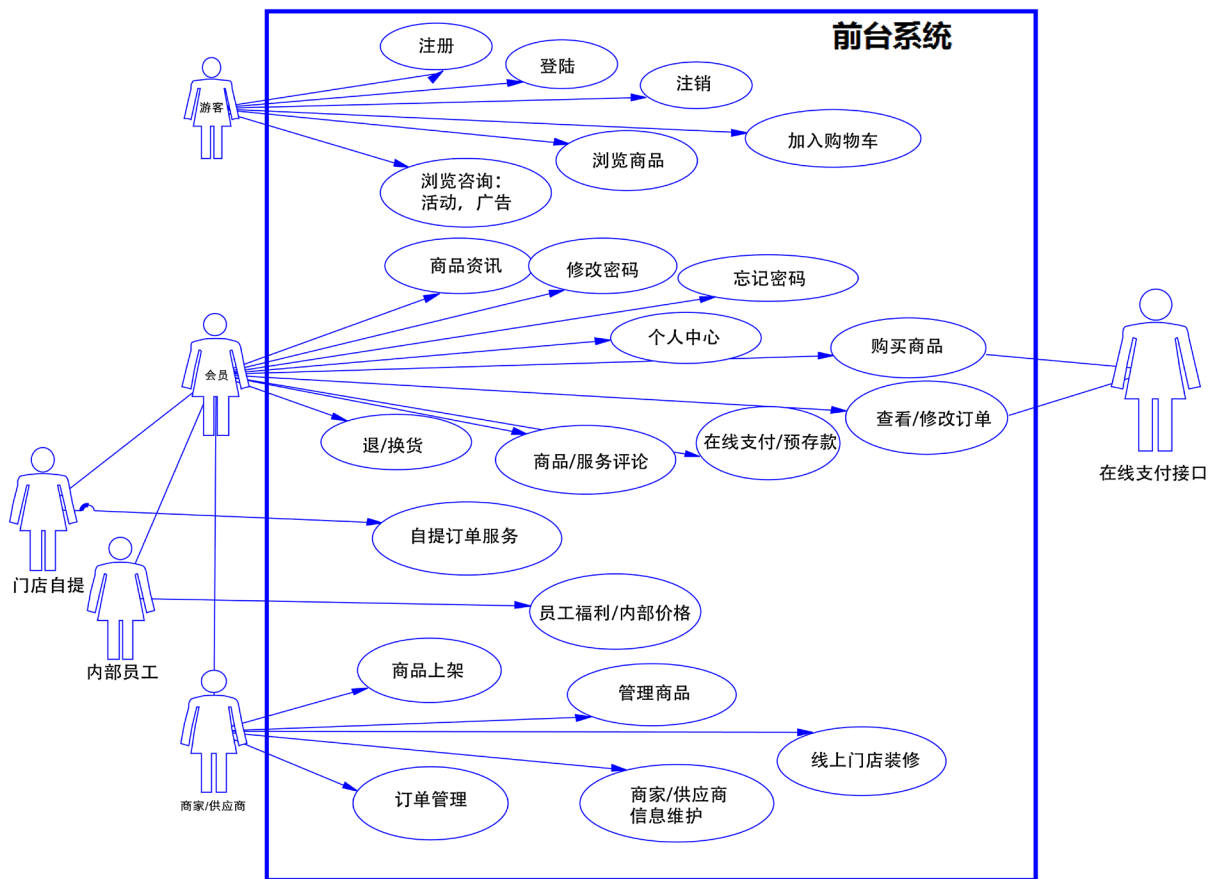


Figure 1. Use case diagram of foreground system

图 1. 前台系统用例图

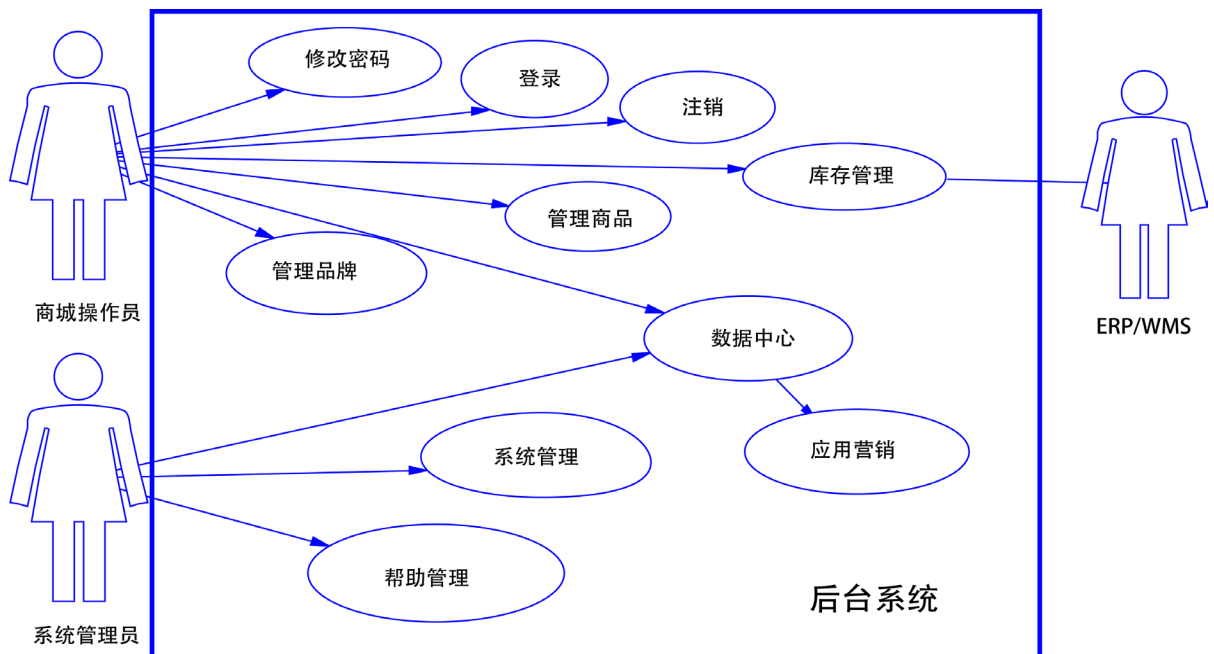


Figure 2. Use case diagram of background system

图 2. 后台系统用例图



Figure 3. Functional architecture diagram of Haitian Online store
图 3. 海田在线商城功能架构图

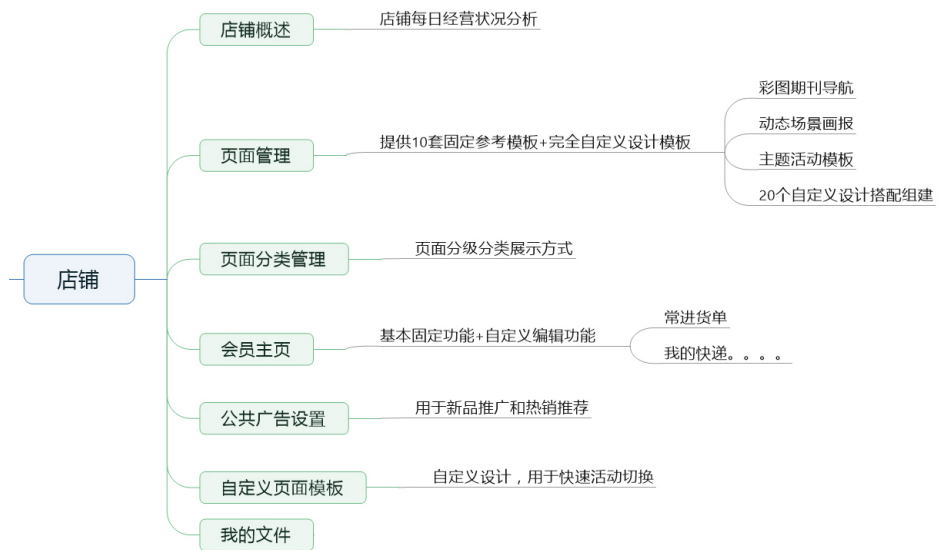


Figure 4. The mind map of store system
图 4. 店铺系统思维导图

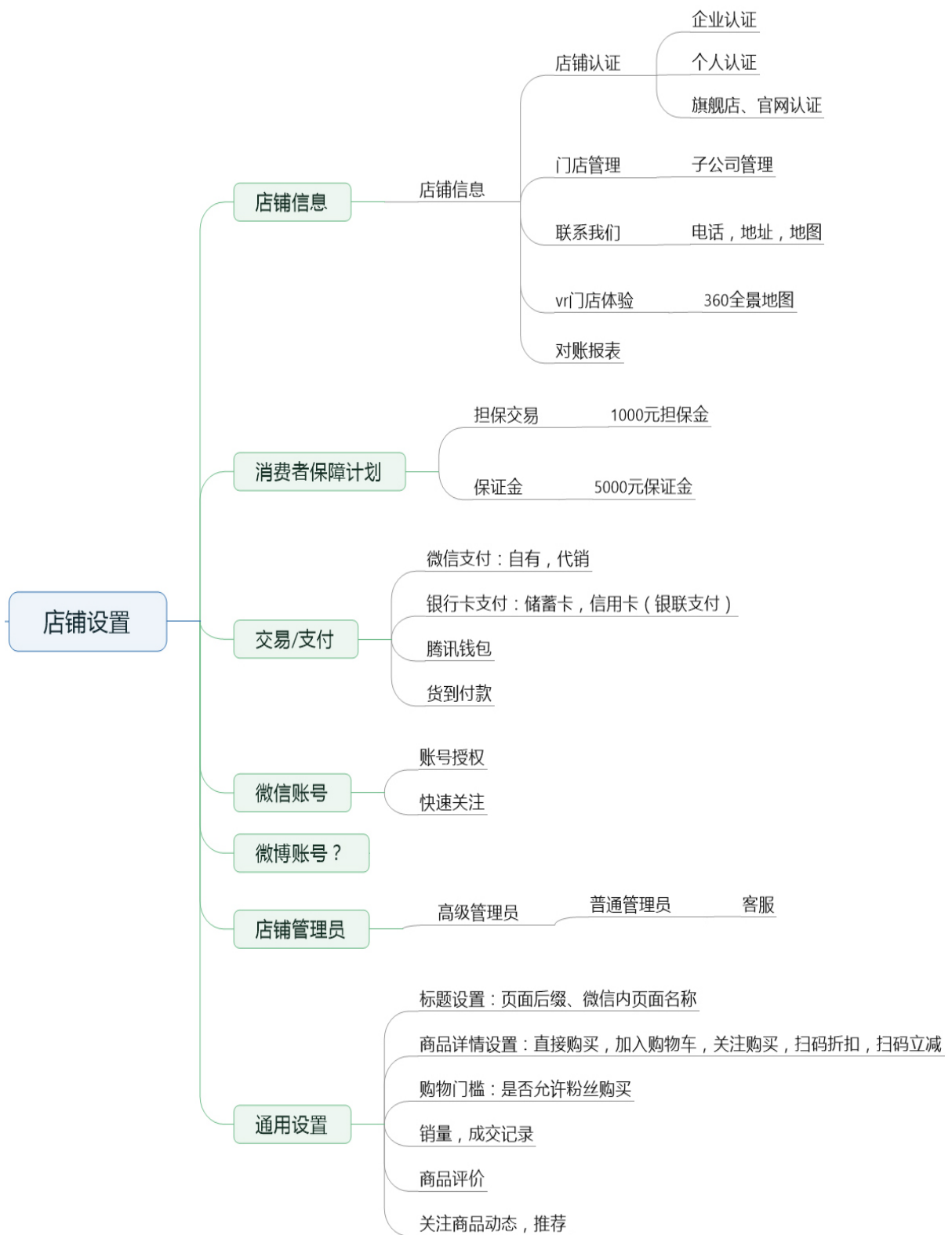


Figure 5. The mind map of store setting
图 5. 店铺设置思维导图

3. 海田在线商城的设计与实现

由于移动互联网的兴起, 海田在线商城需要满足 PC 端、PAD 和智能手机端的跨屏兼容; 满足海田物流产业园 18 个专业市场实现商品的线上管理、线上销售、在线支付等。系统架构图体现的是整个系统的层次性和技术体现, 系统架构图见图 9。

如图 9 所示, 海田商城采用基于 B/S 的三层体系架构模式, 并利用 Dubbo 分布式架构, 将系统把模块拆分并使用接口通信, 降低模块之间的耦合度。可拓展性和分布式部署都相对与传统架构来的灵活。

3.1. 开发技术

3.1.1. Dubbo 分布式解决多用户高并发问题

Dubbo 是阿里巴巴开源的分布式服务框架, 它最大的特点是按照分层的方式来架构, 使用这种方式可以使各个层之间解耦合。透明化的远程方法调用, 就像调用本地方法一样调用远程方法, 只需要简单 API 侵入。架构如图 10 所示。

海田在线商城的开发过程中存在这种情况, 服务提供方(Provider)发布服务以后, 服务消费方(Consumer)发出一次调用请求, 恰好这次由于网络问题调用失败, 虽然可以采用重试这种策略, 但是存在发布服务所在节点出现故障无法提供服务。所以我们搭建 Dubbo 集群解决了多用户同时访问网站某一个服务容易发生冲突的问题, 在出现节点故障无法提供服务的时候, 通过调用其它服务节点来解决, 这就提高了服务的可用性。

3.1.2. Solr 集群实现商品快速搜索

Solr 是基于 Lucene 的全文搜索服务器, 同时对其进行了扩展, 提供了比 Lucene 更为丰富的查询语言同时实现了可配置、可扩展并且优化了查询性能。SolrCloud 是 Solr 提供的分布式搜索方案, 原理是由多台 Solr 服务器集中部署, 共同来完成搜索和所索引库的任务。使用 Solr 集群的方式能更好的解决海田在线商城的多用户高并发、海量数据的搜索需求。

搭建 Solr 集群主要用途是降低单机的处理压力, 将海田商城的搜索服务项目切换到这个集群, 商城



Figure 6. The mind map of commodity system
图 6. 商品系统思维导图

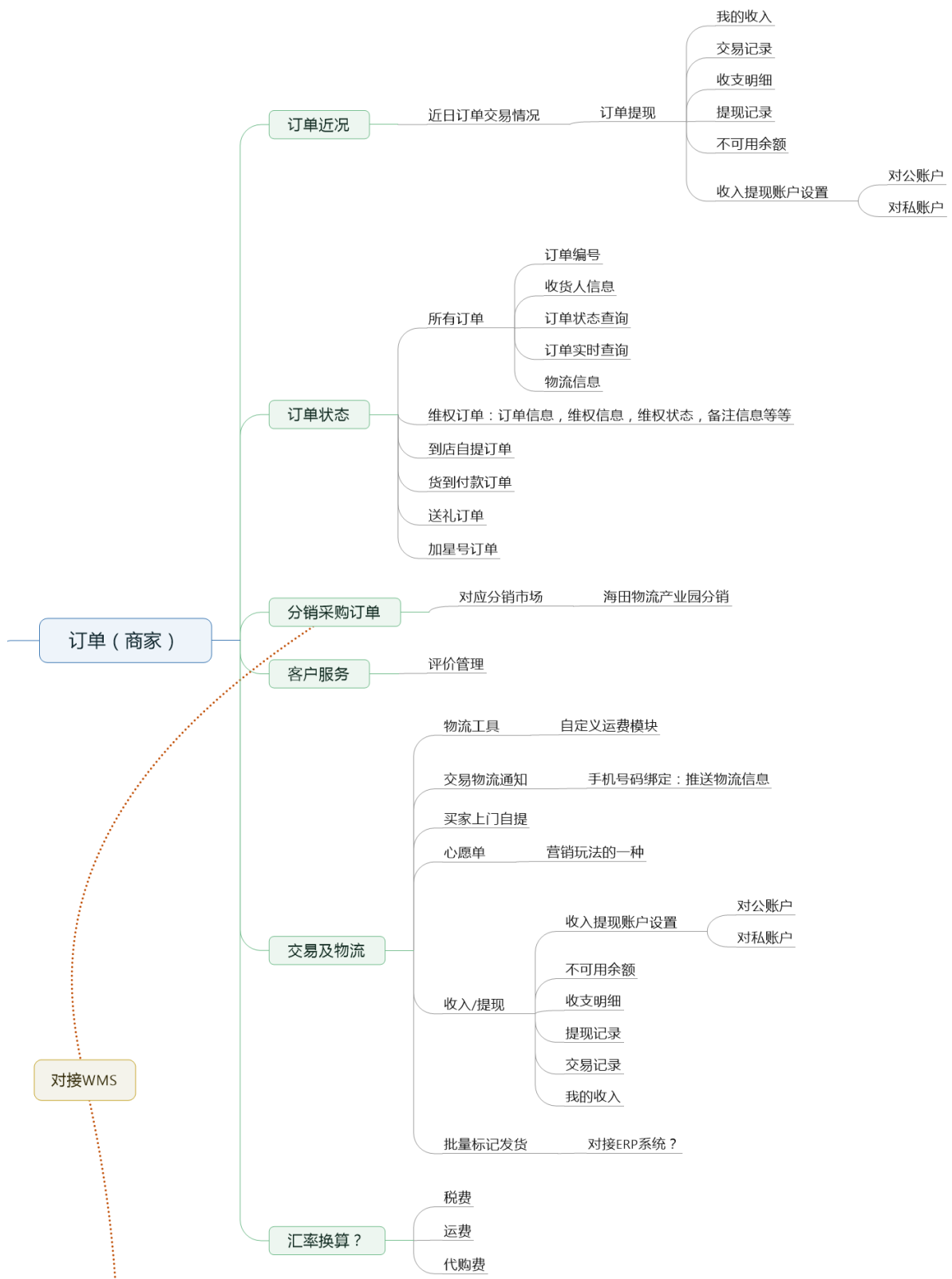


Figure 7. The mind map of order system
图 7. 订单系统思维导图

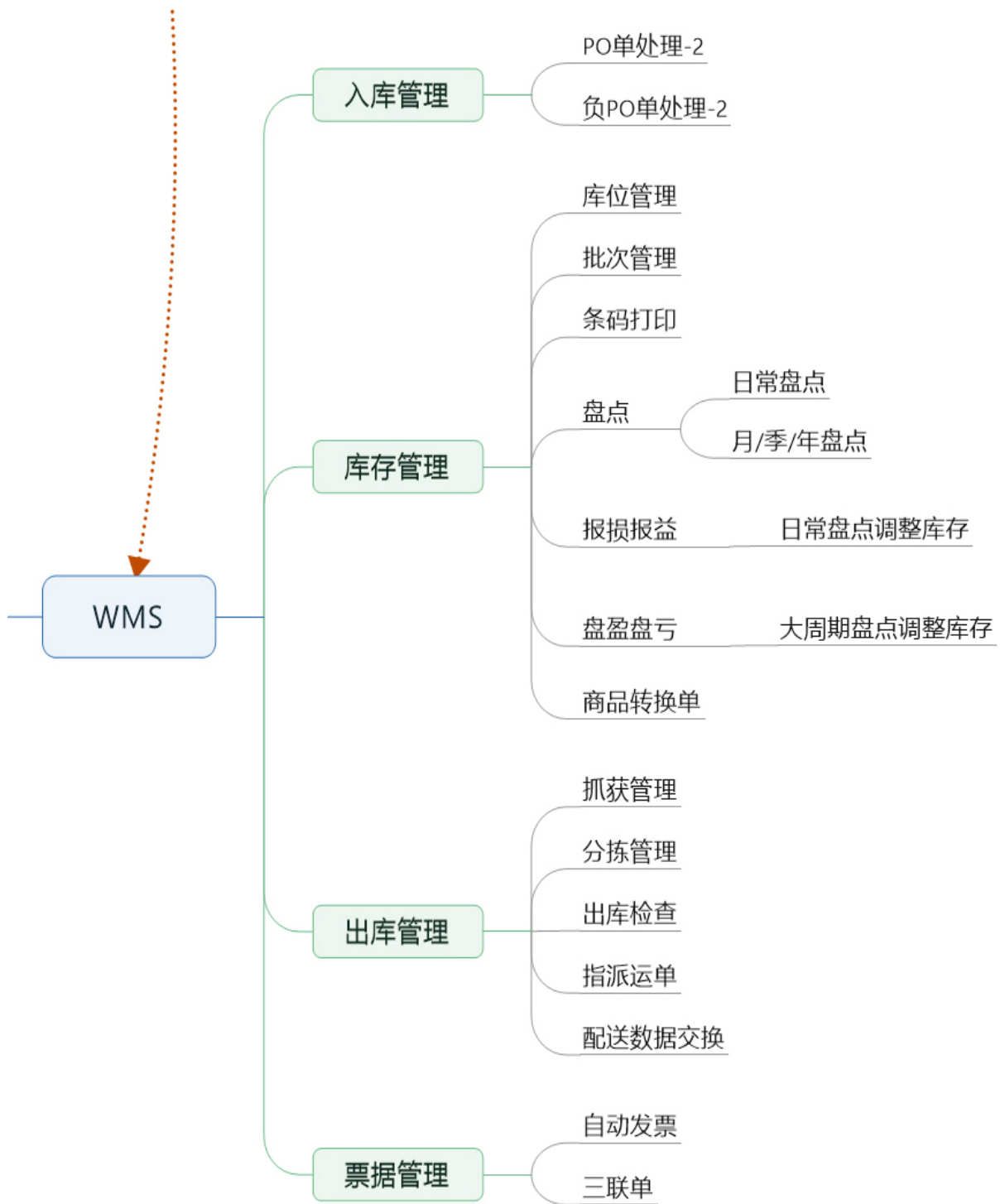


Figure 8. The mind map of WMS
图 8. 仓储系统思维导图

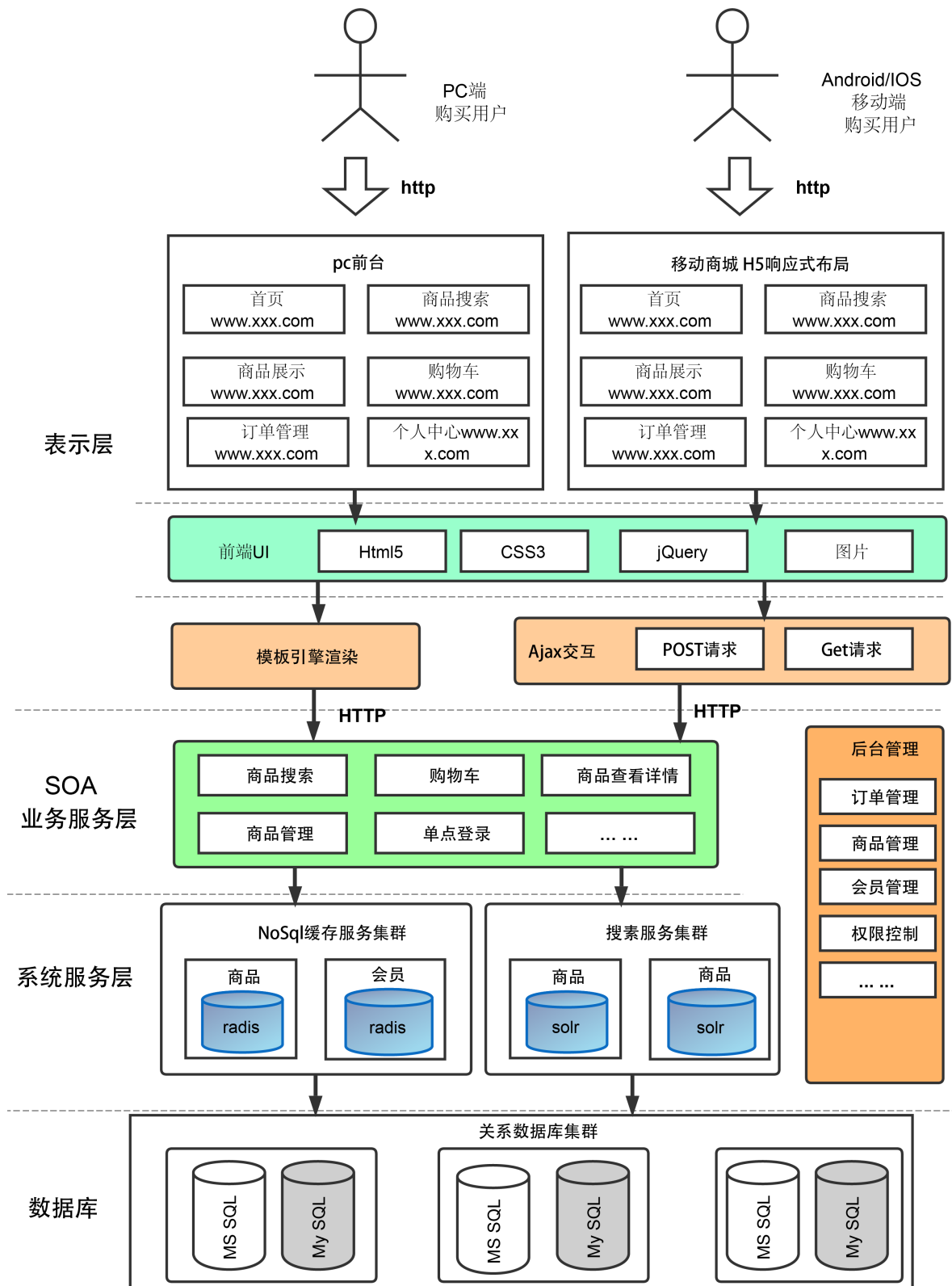


Figure 9. Functional architecture diagram of Haitian Online store
图 9. 海田在线商城系统架构图

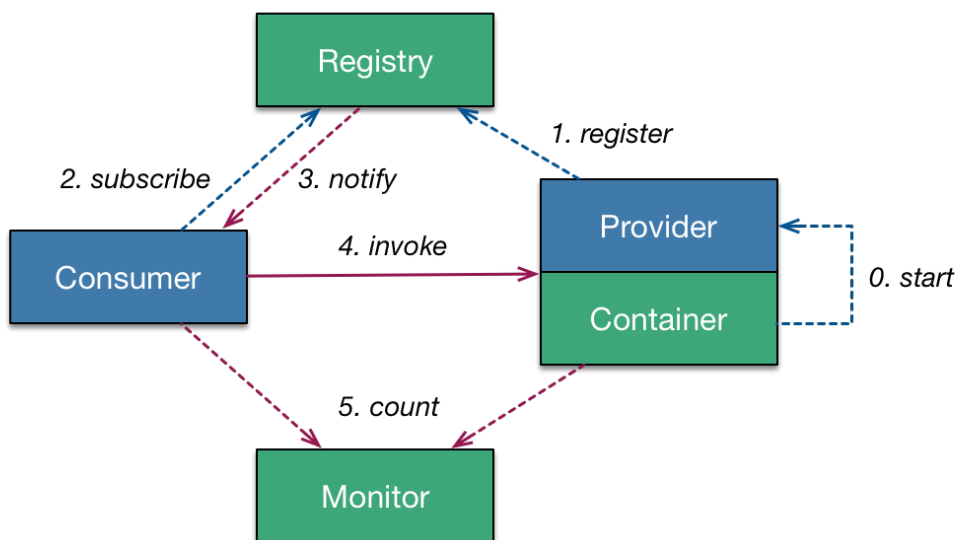


Figure 10. Dubbo Architecture¹
图 10. Dubbo 架构图

中的所有商品信息导入到 Solr 的索引库中，在 java 后台通过 SolrJ 对 Solr 集群进行搜索相关的操作。最终实现对商品快速的全站搜索服务。

3.1.3. Redis 提升数据访问速度

Redis 即为 Remote Dictionary Server (远程数据服务)的缩写是一个开源的，支持网络、可基于内存亦可持久化的日志型、Key-Value 的 Nosql 数据库，并提供 java 等多种语言的 API。

Redis 的主要特点是以内存空间来换数据读取时间，在海田商城的开发中，对于商品信息、商城活动、会员信息等一些短时间内不需要改变数据的，但是又要多次被用户点击访问。Redis 即可缓解这类数据库的压力，降低商城的数据库负荷，海田商城将发布的商品信息保存到 Redis 数据库，用户可以立即查看到，然后将内存中的数据同步到关系型数据库。相对与 Memcached 分布式内存对象缓存系统，Redis 支持存储的 value 类型相对更多，可靠性上比 Memcached 来的强[4]。更适合海田商城这种多位一体的电商平台。

3.2. SSM 框架的研究

SSM 框架整合即为 Spring、SpringMVC、MyBatis 三个开源框架的整合。商城主要是采用 Spring (4.2.4.RELEASE)，MyBatis (3.2.8)进行 SSM 整合。

3.2.1. Spring 框架

Spring 是一个开放源代码、轻量级的 Java 开发框架。通过 Spring 提供的 IoC 容器，我们可以将对象之间的依赖关系交由 Spring 进行控制，避免硬编码所造成的过度程序耦合。在这里举个例子来说明 IoC 的核心思想，支付宝在整个淘宝体系里就是庞大的 IoC 容器，交易双方之外的第三方，提供可靠性可依赖可灵活变更交易方的资源管理中心。我们在设计商城的时候：先设计商家，然后根据商家设计商品，接着根据商品设计买家，最后根据买家设计仓储。这里就出现了一个“依赖”关系：商品依赖商家，买家依赖商品，商品依赖库存。商品、买家、卖家、仓储之间的资源不由使用资源的双方管理，而由不使

¹图片引用自 Dubbo 官方文档，<http://dubbo.io/>.

用资源的第三方即 Spring 的 IoC 容器来管理, 这可以带来很多好处。第一, 资源集中管理, 实现资源的可配置和易管理。第二, 降低了使用资源双方的依赖程度, 也就是我们说的耦合度。这样以后我们在对商品、卖家、买家、仓储的服务上增加或者增强功能模块的时候就不必要在商品、买家、卖家、仓储每个服务去修改, 提高了后期维护的效率。

Spring 还有一个很重要的特性, 就是可以方便集成各种优秀的框架。海田在线商城主要集成了 Dubbo 服务框架、Redis 缓存服务、Solr 搜索服务、ActiveMQ 消息队列服务。Spring 集成海田商城要用到的第三方框架后, 就可以通过简单的配置调用第三方框架提供的 API, 实现商城的业务服务。具体整合及特性如下:

1) 在商城 SOA 业务服务层集成了分布式的 Dubbo 服务框架, 例如使用 Dubbo 实现分布式服务框架来提供相关商品的展示、商品搜索等服务时, 只需在提供这些服务的 web 程序的 SSM 配置文件上加上 Dubbo 的关键配置

```
<dubbo:service interface="提供商品相关服务的接口"ref="商品相关服务的实现类的 bean 的 id"/>
<bean id="商品相关服务的实现类的 bean 的 id" class="商品相关服务的实现类"/>
```

这样就可以把商品服务注册到服务中心, 当要使用这些商品服务的 web 程序, 如商城主页, 就可以透明化的远程方法调用, 就像调用本地方法一样调用远程方法。

2) 在商城的系统服务层整合了用来做缓存的 Redis 缓存库和用来提供搜索服务的 Solr 集群, 使得整个系统数据库的负载大大降低, 提高运行速度。数据库则整合了 MyBatis, 下面详述。

3) Spring 系统中我们还内嵌了 ActiveMQ, ActiveMQ 是强劲的消息总线, 不论是在海田商城分布式开发过程中还是在运维上都扮演着重要的角色, ActiveMQ 的特点主要是, 可以建立网络通信的通道, 进行数据的可靠传送。在传输中保证了数据传输的完整性, 而且能够跨平台传输。

3.2.2. SpringMVC

SpringMVC 属于 Spring 框架的衍生模块, SpringMVC 通过 DispatcherServlet 这个前端控制器(也叫中央调度器, 笔者认为中央调度器更能体现其作用), 来调用 MVC 的三大件: Controller、Model、View。这样就保证 MVC 的每一个组件只与 DispatcherServlet 耦合, 而彼此之间独立运行, 大大降低了程序的耦合性。使得海田商城的设计复合程序的开发之道, 高内聚低耦合的设计原则。

3.2.3. MyBatis 的选用

MyBatis 是集合多种操作型关系数据库的概念和方法, 它是一个强大的数据访问工具和解决方法[5]。对比 Hibernate, MyBatis 是一个半自动框架, Hibernate 则是一个全自动框架, 无法直接维护 SQL。MyBatis 在书写 SQL 的灵活性很好, Hibernate 就不行, 会比较麻烦。本文使用 MyBatis 作为海田在线商城这种多变性的项目, Hibernate 比较适合稳定的项目。而且 MyBatis 速度相对于 Hibernate 的速度要快些。

3.3. 功能页面结构实现

3.3.1. 前端系统架的搭建

前端系统的搭建(如图 11 所示)主要是实现前后端分离开发的, 只在数据库层面进行交互上, 这样前端能够缓解多用户、高并发的问题。有利于整个平台的性能能够更高效的运行。后台则通过发布服务, 供 PC 端和移动端公用, 提高了代码的复用度, 降低了系统之间耦合度。

3.3.2. 前端页面的实现

基于 SSM 框架的后台系统, 前台系统主要实现了功能模块页面有: 首页信息模块、商品展示信息模块、购物车信息模块、商品展示信息模块、用户管理模块、订单管理模块等 12 个模块。所有模块都能够实现 PC 端(如图 12 所示)、PAD 端和智能手机端(如图 13 所示)的分辨率自适应, 满足绝大部分的用户需求。

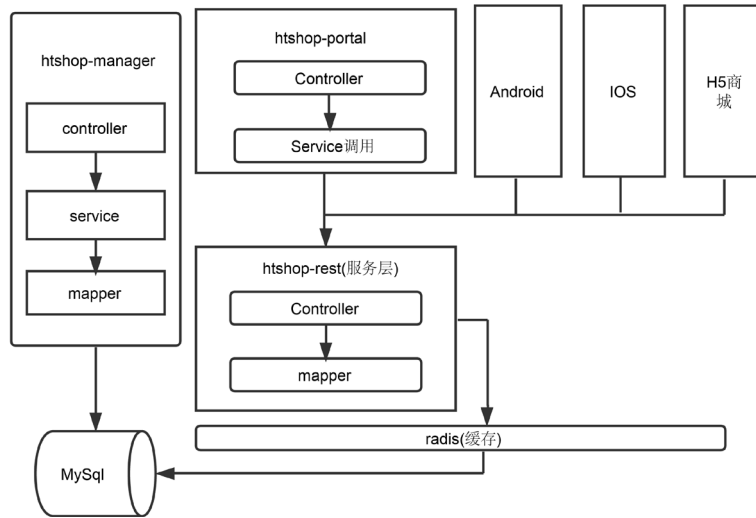


Figure 11. Functional architecture diagram of foreground system
图 11. 前端系统架构图



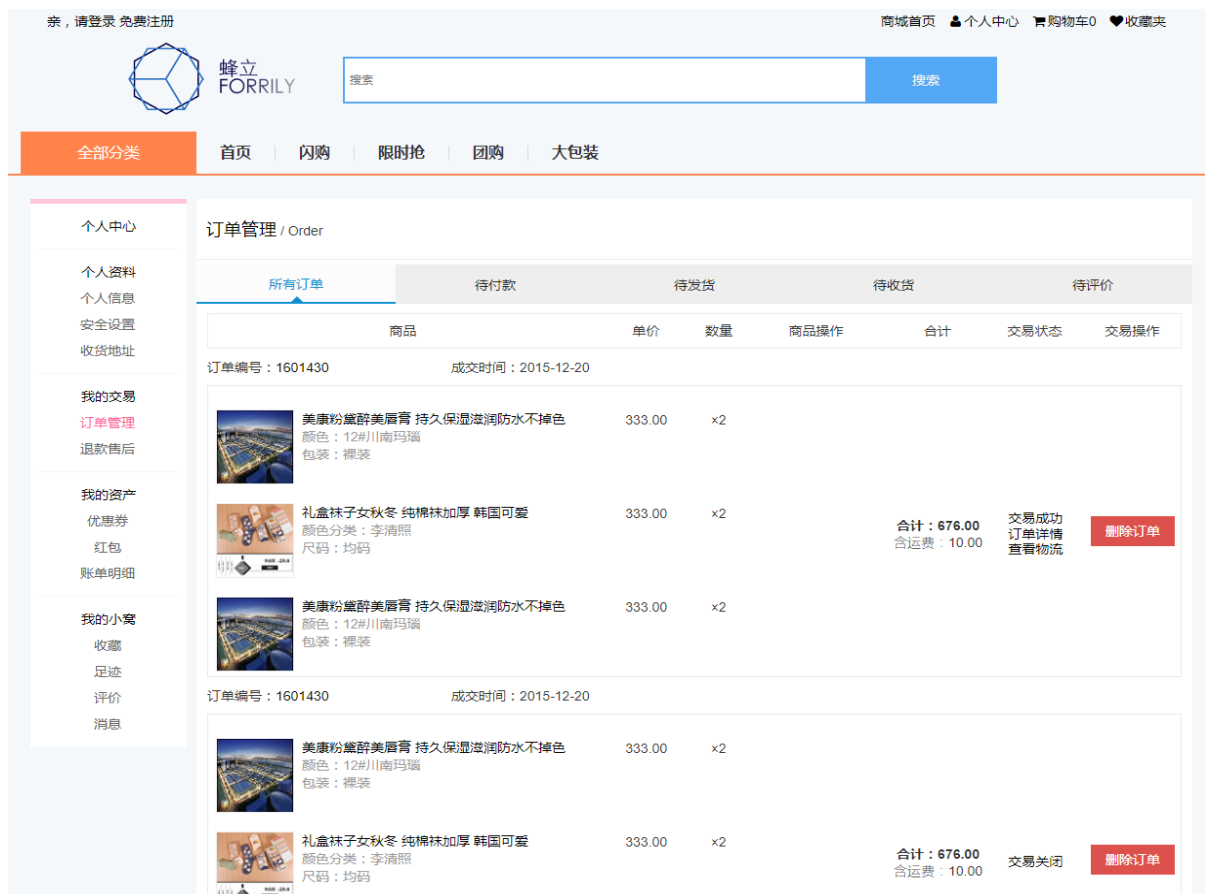


Figure 12. PC-side page structure
图 12. 海田在线商城 PC 端页面结构

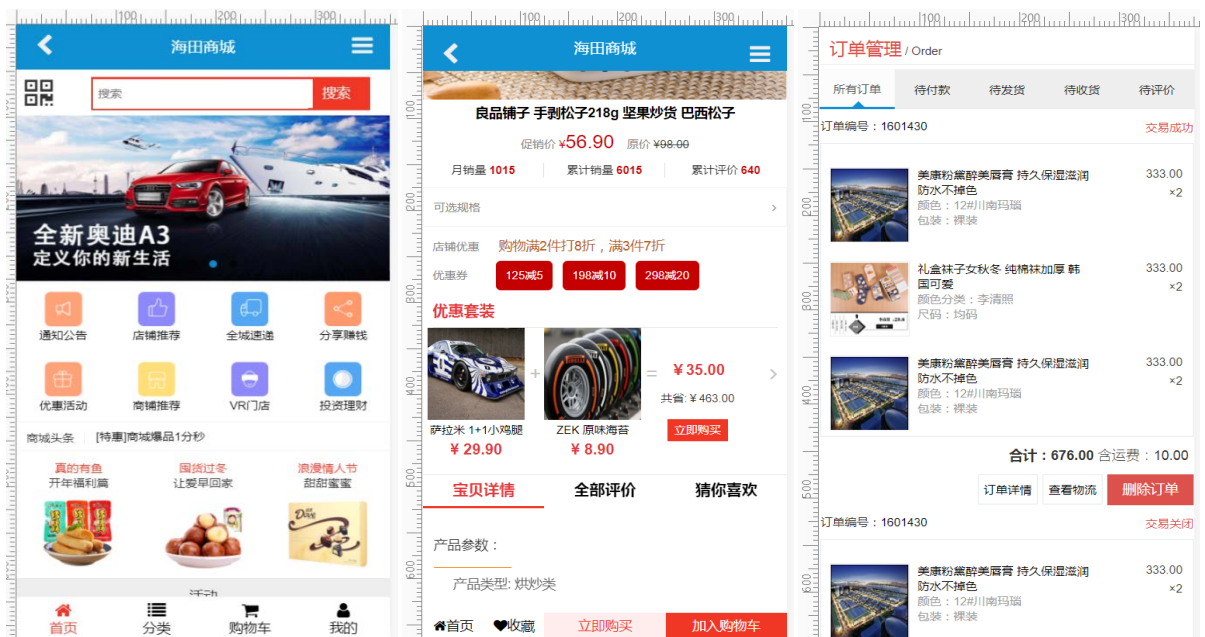


Figure 13. Smart mobile phone Page Structure
图 13. 海田在线商城智能手机端页面结构

Table 1. Data sheet list summary
表 1. 数据表清单内容汇总

| 表名 | 描述 |
|---------------------------|-------------|
| zjhtshop_content | 内容管理表 |
| zjhtshop_content_category | 内容分类表 |
| zjhtshop_item | 商品详情表 |
| zjhtshop_item_cat | 商品分类表 |
| zjhtshop_item_desc | 商品描述表 |
| zjhtshop_item_param | 商品规则参数表 |
| zjhtshop_item_param_item | 商品规格和商品的关系表 |
| zjhtshop_order | 订单详情表 |
| zjhtshop_order_item | 订单商品表 |
| zjhtshop_order_shipping | 订单收货详情表 |
| zjhtshop_user | 用户表 |

内容管理表/zjhtshop_content

| 名称 | 注释 | 数据类型 |
|-------------|--------|--------------|
| id | 内容编号 | bigint(20) |
| category_id | 内容类目ID | bigint(20) |
| title | 内容标题 | varchar(200) |
| sub_title | 子标题 | varchar(100) |
| title_desc | 标题描述 | varchar(500) |
| url | 链接 | varchar(500) |
| pic | 图片绝对路径 | varchar(300) |
| pic2 | 图片2 | varchar(300) |
| content | 内容 | text COMMENT |
| created | 创建时间 | datetime |
| updated | 更新时间 | datetime |

内容分类表/zjhtshop_content_category

| 名称 | 注释 | 数据类型 |
|------------|--------|-------------|
| id | 类目ID | bigint(20) |
| parent_id | 父类目ID | bigint(20) |
| name | 分类名称 | varchar(50) |
| status | 状态 | int(1) |
| sort_order | 排列序号 | int(4) |
| is_parent | 该类目是否为 | tinyint(1) |
| created | 创建时间 | datetime |
| updated | 更新时间 | datetime |

商品分类表/zjhtshop_item_cat

| 名称 | 注释 | 数据类型 |
|------------|--------|-------------|
| id | 类目ID | bigint(20) |
| parent_id | 父类目ID | bigint(20) |
| name | 类目名称 | varchar(50) |
| status | 状态 | int(1) |
| sort_order | 排列序号 | int(4) |
| is_parent | 该类目是否为 | tinyint(1) |
| created | 创建时间 | datetime |
| updated | 创建时间 | datetime |

商品描述表/zjhtshop_item_desc

| 名称 | 注释 | 数据类型 |
|-----------|------|------------|
| item_id | 商品ID | bigint(20) |
| item_desc | 商品描述 | text |
| created | 创建时间 | datetime |
| updated | 更新时间 | datetime |

商品详情表/zjhtshop_item

| 名称 | 注释 | 数据类型 |
|----------|-------|--------------|
| id | 商品id | bigint(20) |
| title | 商品标题 | varchar(100) |
| sell_poi | 商品卖点 | varchar(500) |
| price | 商品价格 | bigint(20) |
| num | 库存数量 | int(10) |
| barcode | 商品条形码 | varchar(30) |
| image | 商品图片 | varchar(500) |
| cid | 所属类目 | bigint(10) |
| status | 商品状态 | tinyint(4) |
| created | 创建时间 | datetime |
| updated | 更新时间 | datetime |

商品规格参数表/zjhtshop_item_param

| 名称 | 注释 | 数据类型 |
|----------|----------|------------|
| id | 商品规格参数编号 | bigint(20) |
| item_cat | 商品类目ID | bigint(20) |
| param_da | 参数数据 | text |
| created | 创建时间 | datetime |
| updated | 更新时间 | datetime |

商品规格和商品关系表/zjhtshop_item_param_item

| 名称 | 注释 | 数据类型 |
|----------|-------------|------------|
| id | 商品规格和商品关系id | bigint(20) |
| item_id | 商品ID | bigint(20) |
| param_da | 参数数据 | text |
| created | 创建时间 | datetime D |
| updated | 更新时间 | datetime D |

用户表/zjhtshop_user

| 名称 | 注释 | 数据类型 |
|----------|-------|-------------|
| id | 用户id | bigint(20) |
| username | 用户名 | varchar(50) |
| password | 密码 | varchar(32) |
| phone | 注册手机号 | varchar(20) |
| email | 注册邮箱 | varchar(50) |
| created | 创建时间 | datetime |
| updated | 更新时间 | datetime |

订单详情表/zjhtshop_order

| 名称 | 注释 | 数据类型 |
|---------------|----------|--------------|
| order_id | 订单id | varchar(50) |
| payment | 实付金额 | varchar(50) |
| payment_type | 支付类型 | int(2) |
| post_fee | 邮费 | varchar(50) |
| status | 状态 | int(10) |
| create_time | 订单创建时间 | datetime |
| update_time | 订单更新时间 | datetime |
| payment_time | 付款时间 | datetime |
| consign_time | 发货时间 | datetime |
| end_time | 交易完成时间 | datetime |
| close_time | 交易关闭时间 | datetime |
| shipping_name | 物流名称 | varchar(20) |
| shipping_no | 物流单号 | varchar(20) |
| user_id | 用户id | bigint(20) |
| buyer_me | 买家留言 | varchar(100) |
| buyer_ni | 买家昵称 | varchar(50) |
| buyer_ra | 买家是否已经评价 | int(2) |

订单商品表/zjhtshop_order_item

| 名称 | 注释 | 数据类型 |
|----------|--------|--------------|
| id | 订单商品id | varchar(20) |
| item_id | 商品id | varchar(50) |
| order_id | 订单id | varchar(50) |
| num | 商品购买数量 | int(10) |
| title | 商品标题 | varchar(200) |
| price | 商品单价 | bigint(50) D |
| total_fe | 商品总金额 | bigint(50) D |
| pic_path | 商品图片地址 | varchar(200) |

订单收货详情表/zjhtshop_order_shipping

| 名称 | 注释 | 数据类型 |
|----------|-------|--------------|
| order_id | 订单ID | varchar(50) |
| receiver | 收货人全名 | varchar(20) |
| receiver | 固定电话 | varchar(20) |
| receiver | 移动电话 | varchar(30) |
| receiver | 省份 | varchar(10) |
| receiver | 城市 | varchar(10) |
| receiver | 区/县 | varchar(20) |
| receiver | 收货地址 | varchar(200) |
| receiver | 邮政编码 | varchar(6) |
| created | 创建时间 | datetime |
| updated | 更新时间 | datetime |

Figure 14. Data table field design

图 14. 数据表字段设计

3.4. 数据库设计

海田在线商场依照初定的项目需求设计了表 1 所示数据表。

海田在线商城的数据表字段及类型的具体设计如图 14 所示。

4. 结束语

本文对于海田在线商城的设计进行了初步实现, 采用 Spring、SpringMVC 和 MyBatis 框架为基础, Spring 作为整个系统的控制中心, 这样可以使得系统在整合上更符合程序的设计之道, 降低了整个海田商城系统的复杂度。Dubbo、Redis、SolrCloud 技术的引入能够解决在线商城的海量数据、多用户高并发的问題, 在移动互联网不断更新的时代里能够更好的兼容和内嵌新技术框架。一个好的平台需要不断的完善和改进, 其中包括了问题的解决, 例如本项目在开发中解决 js 跨域访问的问题应使用 jsonp、跨站请求伪造(CSRF 攻击)的解决方法为设置验证码验证, 增加 Token 验证等[6]。这种基于 SSM 三大框架整合开发企业型的商城还是很受用的, 分布式开发能够灵活的分布和部署移动端或者 PC 的服务器。对于商城的后续完善还有待进一步研究。

基金项目

2017 年“攀登计划”广东大学生科技创新培育专项资金资助(项目编号: pdjh2017b0983)。

参考文献 (References)

- [1] 陈伟民, 赵薇. 以“互联网+”打造“智慧物流” [N]. 今日路桥, 2016-11-26.
- [2] 梁丹. UML 在面向对象的关系数据库中的应用[J]. 山西科技, 2009(3): 34-35.
- [3] 杨开振, 周吉文, 梁华辉, 谭茂华. Java EE 互联网轻量级框架整合开发 SSM 框架(Spring MVC + Spring + MyBatis)和 Redis 实现[M]. 北京: 电子工业出版社, 2017.
- [4] 王心妍. Memcached 和 Redis 在高速缓存方面的应用[J]. 无线互联科技, 2012(9): 8-9.
- [5] 徐雯, 高建华. 基于 Spring MVC 及 MyBatis 的 Web 应用框架研究[J]. 微型电脑应用, 2012, 28(7): 1-4.
- [6] 吴翰清. 白帽子讲 Web 安全(纪念版) [M]. 北京: 电子工业出版社, 2017.

知网检索的两种方式:

1. 打开知网页面 <http://kns.cnki.net/kns/brief/result.aspx?dbPrefix=WWJD>
下拉列表框选择: [ISSN], 输入期刊 ISSN: 2325-2286, 即可查询
2. 打开知网首页 <http://cnki.net/>
左侧“国际文献总库”进入, 输入文章标题, 即可查询

投稿请点击: <http://www.hanspub.org/Submission.aspx>
期刊邮箱: sea@hanspub.org