

硕士研究生学位论文

|  |  |
| --- | --- |
| 题目： | 电子商务促销系统 |
|  | 的设计与实现 |

|  |  |
| --- | --- |
| 姓 名： | 龙利剑 |
| 学 号： | 1201220801 |
| 院 系： | 软件与微电子学院 |
| 专 业： | 软件工程 |
| 研究方向： | 嵌入式系统设计 |
| 导师姓名： | 吴中海、张齐勋 |

二〇一五年八月

版权声明

任何收存和保管本论文各种版本的单位和个人，未经本论文作者同意，不得将本论文转借他人，亦不得随意复制、抄录、拍照或以任何方式传播。否则，引起有碍作者著作权之问题，将可能承担法律责任。

# 摘要

本论文基于某电商平台，旨在开发出一个能够高效支持运营团队发布营销活动（满减、满赠等）而开发的电商促销系统。

整个流程为：运营人员通过商品管理系统来发布货物，销售管理系统来发布促销活动商品。用户通过购买相关促销商品来享受促销折扣。整个过程使用WEB技术来实现促销商品的发布（包括促销规则的实现，商品参加多个促销活动成本的控制等）；通过用户注册数据和行为数据，使用机器学习推荐算法推荐用户最合适的促销活动。

论文从项目背景出发，分别介绍了该系统开发的价值和意义。采用NGINX + PHP + MySQL和推荐算法等项目相关技术构建这个促销系统。用NGINX作为WEB服务器，处理用请求。使用PHP脚本语言设计实现后台促销业务逻辑，使用MySQL作为数据库存储业务数据。使用离线式的机器学习算法利用HADOOP技术用户可能关注的促销活动。并分别介绍了各模块的详细设计。本论文阐述了系统的需求，并同时介绍了网站的设计及其实现，最后，针对本系统进行了分析和评价，总结全文并阐述今后工作的展望。

关键词：促销系统，电子商务，推荐算法，机器学习，WEB技术

# ABSTRACT

In this paper, based on a business platform, designed to support the development of an operations team to efficiently release marketing activities (full cut, full of gifts, etc.) and the development of electronic business promotion system.

Whole process: operations staff by merchandise management system to publish goods, sales management system to publish promotional activities merchandise. Users enjoy promotional discounts by purchasing related promotional merchandise. WEB entire process uses technology to achieve publish promotional merchandise (including the implementation of more promotional activities to participate in commodity cost control and other promotional rules); through user registration data and behavioral data, using machine learning algorithms recommend users recommend the most appropriate promotional activities.

Papers from the project background, introduced the value and significance of the system development, also introduced NGINX + PHP + MySQL and other projects related recommendation algorithm technology. And introduced the detailed design of each module. This paper describes the requirements of the system, and also introduces the design and realization of the site, finally, for the system were analyzed and evaluated, concluding remarks and future prospects elaborate work.

KEY WORDS: Promotion system,Machine learning

# 目录

[第一章 绪论 5](#_Toc429473992)

[1.1 项目背景和意义 5](#_Toc429473993)

[1.2 总体目标 6](#_Toc429473994)

[1.3 本文结构 6](#_Toc429473995)

[第二章 系统相关技术简述 8](#_Toc429473996)

[2.1 NGINX 简介 8](#_Toc429473998)

[2.2 PHP 简介 8](#_Toc429473999)

[2.2.1 基本概念 8](#_Toc429474000)

[2.2.2 应用领域 9](#_Toc429474001)

[2.2.3 基本工作原理 9](#_Toc429474002)

[2.2.4 选择PHP的优势 10](#_Toc429474003)

[2.4 促销推荐简介 11](#_Toc429474004)

[2.4.1 促销商品热度数据 11](#_Toc429474005)

[2.4.2 促销系统推荐系统 12](#_Toc429474006)

[2.5 促销系统运行流程 13](#_Toc429474007)

[2.6 Hadoop 简介 13](#_Toc429474008)

[2.7 本章小结 14](#_Toc429474009)

[第三章 系统需求分析和概要设计 16](#_Toc429474010)

[3.1 系统需求分析 16](#_Toc429474011)

[3.1.1 功能性需求 16](#_Toc429474012)

[3.1.2 非功能性需求 16](#_Toc429474013)

[3.2 概要设计 17](#_Toc429474014)

[3.2.1 商品管理系统 17](#_Toc429474015)

[3.2.2 商品销售管理系统 17](#_Toc429474016)

[3.2.3 订单系统 18](#_Toc429474017)

[3.3 促销系统 19](#_Toc429474018)

[3.3.1 促销活动规则 19](#_Toc429474019)

[3.3.2 促销活动创建 20](#_Toc429474020)

[3.3.3 促销活动均摊逻辑设计 20](#_Toc429474021)

[3.3.4 促销活动热度数据 20](#_Toc429474022)

[3.3.5 促销活动推荐 20](#_Toc429474023)

[3.4 本章小结 20](#_Toc429474024)

[第四章 系统详细设计 22](#_Toc429474025)

[4.1 商品发布模块 22](#_Toc429474026)

[4.2 订单系统模块 22](#_Toc429474027)

[4.3 促销系统模块 22](#_Toc429474028)

[4.3.1 促销系统发布模块 22](#_Toc429474033)

[4.3.2 促销系统推荐模块 22](#_Toc429474034)

[4.4 本章小结 22](#_Toc429474035)

[第五章 数据库设计 23](#_Toc429474036)

[5.1 数据模型 23](#_Toc429474037)

[5.2 数据模型详细描述 23](#_Toc429474038)

[5.3 本章小结 26](#_Toc429474039)

[第六章 系统的实现 27](#_Toc429474040)

[6.1 发布促销活动 27](#_Toc429474041)

[6.2 促销显示 27](#_Toc429474042)

[6.3 促销结算 27](#_Toc429474043)

[6.4 促销商品退货计算 27](#_Toc429474044)

[6.5 促销推荐 27](#_Toc429474045)

[6.6 本章小结 27](#_Toc429474046)

[第七章 总结与展望 28](#_Toc429474047)

[7.1 总结 28](#_Toc429474048)

[7.2 展望 28](#_Toc429474049)

[参考文献 30](#_Toc429474050)

[附录A 附录示例 31](#_Toc429474051)

[致谢 32](#_Toc429474052)

[北京大学学位论文原创性声明和使用授权说明 32](#_Toc429474053)

# 绪论

* 1. 项目背景和意义

时下，伴随着互联网信息技术的高速的发展。利用互联网信息技术，整合线上线下资源，高效的为互联网用户提供优质的服务，是大势所趋。电子商务就是一种目前被普遍接受认同的方式。近几年国内的电子商务层出不穷，经过过年的发展，人们正在享受信息化时代的浪潮。

北京某公司是中国著名的农产品交易平台，为中国上千家餐饮连锁店提供优质的蔬菜水果售卖服务。该公司目前主要采用跟农户对接，把采集过来的农产品对接到菜市场，餐饮店等。运营互联网技术打造这样的一个B2C平台。为农民增收减负，解决农民销售农产品困难，同时为餐饮行业提供优质安全的产品，免去餐饮工作人员每天凌晨去菜市场，采购食材之苦。

本项目就是在这样的背景下产品的一个基于电子商务平台而开发的促销系统，旨在提高公司销售额，提高市场份额，同时给用户以最优的价格提供给用户最优质的服务。促销系统作为每一个电子商务的重要组成部分，是公司运营发布活动的重要操作窗口，目前市面上各家的促销系统支持的活动形式各异。同时因为竞争关系，每家电商在实现自己的促销系统方式上也不尽相同。但是总的目的都是一样的就是为了支持公司能够应对友商提高一个有力的竞争工具，来发布促销策略。让用户享受到更优惠的价格。促销活动可以定义为：给什么用户提供什么方式的优惠活动。不管什么促销总体上都是这样要支持：1.能够制定活动范围，2.发布什么样子的活动。

本项目在分析了产品的需求之上，设计了自己的促销系统技术框架。不仅实现了运营团队发布促销活动的任务，同时引入了当前流行的大数据技术并结合现有系统的DM(数据挖掘)标签管理系统使得运营人员发布的促销活动能够在用户端有效的查看到用户关注的促销活动。这大大减少了用户因为信息过载而未能发现它感兴趣的促销活动的可能性。从实际的效果来看，开发促销系统，根据公司BI部门的统计。再促销活动期间，销售额度提高200%相对于没有促销活动。同时添加促销推荐模块整体的销售额度提高在这基础上再提高15%。相比于发传单之类的促销活动，使用信息技术平台不仅仅降低了人员成本，还能在促销活动中得用户的反馈实现更精细化的运营。具有更大的价值和收益。

* 1. 总体目标

促销系统因商品运营情况有时要减少滞销商品，有时因友商的一些促销活动，要能及时的提供一些促销促销方案。同时降低人力运营成本等总结出促销系统的总体目标为：

1. 能够低成本运营
2. 能够灵活的发布促销活动
3. 发布的活动能够方便用户知道
4. 能够灵活的扩展出新的促销活动，应对多变的促销方案
5. 能够对成本有很好的把控，要对促销的商品进行限购
6. 为了防止亏损运营，一个商品不能参加多个促销活动

通过对公司需求的分析，以及参考亚马逊、京东等电商的促销活动策略制定了一系列的促销活动规则。每个促销活动都有相应的一系列的活动规则结合。来规避亏损运营，同时使用用户购买记录，我的关注纪录，用户注册信息来给用户推荐相关的促销活动。

考虑现有框架和技术情况使用NGINX + PHP + MySQL的技术框架来设计发布促销活动。采用这样的框架不仅仅是因为开源免费。更多的是因为这些技术相对成熟可靠。对用户体验有一定的保障。同时他们相对灵活，高效配置也比较方便。

促销推荐系统的设计采用离线的方式算好。因为一般商户不会经常购买可能一天也就几次。正常一天购买一次。对时间要求不是很高。同时推荐算法要大量的计算。在用户购买的商品的时候，正确的做法是用户—登录就已经把用户可能相关的感兴趣的促销商品已经算好。这样用户—登录我们就能把他可能感兴趣的促销活动以最快的速度呈现出来。

同时页面的呈现部分，当用户的购物车还差多少可以参与什么促销活动，我们可以提示再买什么，再买多少免除运费。或者赠送某个商品活优惠券信息等。

* 1. 本文结构

本文主要讨论了时下，电子商务的应用背景，以及从工程的角度讲述开发电子商务促销系统的对企业营销方案的支持程度。并针对需求情况，描述了整个项目的设计框架和部分详细设计细节。最后对整个项目进行总结和展望。

本论文各章具体细节如下：

第一章 简要的阐述了项目开发背景及其意义，并根据需求提出项目目标要求。

第二章 论述了相关技术的在国内外的发展状况

第三章 介绍了系统需求设计、概要设计

第四章 介绍了系统各模块的详细设计

第五章 介绍了系统数据库设计

第六章 描述相关功能的测试情况，及该促销系统的相关指标

第七章 总结全文、展望未来，为后续研究开发奠定基础

# 系统相关技术简述

本项目采用NGINX + PHP + MySQL 的技术架构来实现促销应用的实现。使用nginx作为服务器处理用户请求，使用PHP来生产动态网页数据，使用MySQL来保存业务数据。使用Python来处理用户注册信息，购买记录等日志信息。使用Hadoop分布式计算平台分析用数据。项目中各个关键组件选择的开发环境如下：

表 2-1 开发环境相关环境配置情况

|  |  |
| --- | --- |
| 名称 | 版本信息 |
| OS | Ubuntu 64 |
| NGINX | nginx/1.4.6 (Ubuntu)。 |
| PHP | PHP 5.5.9-1ubuntu4.9 |
| MySQL | 5.5.41 |
| Python | 2.7.6 |
| Hadoop | 2.6.0 |



## NGINX 简介

Nginx聚焦于高性能，高并发和低内存消耗问题。并且具有多种web服务器功能特性：负载均衡，缓存，访问控制，带宽控制，以及高效整合各种应用的能力，这些特性使nginx很适合于现代网站架构。相对于apache服务器而言nginx相对比较年轻。但是他本身的处理高并发能力，是我们电子商务网站看重的一个重要特性。

* 1. PHP 简介
     1. 基本概念

PHP原始为Personal Home Page (个人主页) 的缩写，现已经正式更名为“PHP:Hypertext Proprocessor”（超文本预处理器）。PHP于1994年由Rasmus Lerdorf创建，后几经发展在WEB开发中非常重要的地位。它主要包含的特点是：

1. PHP语法混合了JAVA, C, Perl以及PHP自创新的语法
2. PHP可以比Perl或者CGI更快的执行动态网页，动态页面生成的性能，相比其他编程语言。PHP是将脚步程序嵌套在html文档中去执行的，执行效率比完全生成html的CGI要快速的多。
3. 支持的操作系统多，支持几乎所有主流的操作系统和数据库
4. 对于某些高性能的模块可以使用C、C++进行程序扩展
   * 1. 应用领域

PHP正吸引着越来越多的WEB开发人员。PHP给企业和用户带来众多好处。根据相关数据统计，全世界有超过2200万的网站和1.5万家公司在使用PHP语言，包括百度、雅虎、Google、YouTube、Digg等著名网站，也包括汉莎航空电子订票系统、德意志银行的网上银行、华尔街在线的金融信息发布系统等，甚至军队系统这类苛刻的环境也选择使用PHP语言。PHP也是企业用来构建服务导向型、创造和混合WEB于一体新一代综合性商业所使用的语言，成为开源商业应用发展的方向。

1. PHP能够根据网站的访问者客户端的语言设置为访问者提供本地化的服务，自动的地以访问者的母语或习惯语言提供页面。
2. PHP能够很容易地创建flash、pdf等多媒体文件。
3. PHP能都有效地支持加密，并且支持多种数据库服务器。因此，[PHP](http://www.phpdo.net/)可用于企业内部管理系统的开发、B/S架构的电子商务系统的开发、公共网站的卡发等。

PHP主要擅长的领域**：**

1. 内容管理系统（CMS）：用于管理新闻、资料数据等。包括前台和后台。典型产品如DedeCms等。适合于个人网站及一般商业网站的应用。
2. 论坛系统（Forum）：是一个支持用户间传递和共享信息的交流平台。典型产品如Discuz等。Discuz在稳定、负载、安全等方面都是领先的。
3. 电子商务系统（e-Business）:典型产品如shopEX等。
   * 1. 基本工作原理

和很多WEB架构一样，都是被动的处理用户请求。一个回话的过程一定是由客户端先发起的，然后客户的请求通过网络传输到服务器上，比如我们选择nginx。然后服务器根据用户的请求分析用户要访问的数据由nginx的work进程去处理用户的请求。服务器进行解析用户的请求资源，如果碰到php服务器会调用php-fpm进程去叫PHP的解释去解释执行用户请求的数据代码时什么，如果碰到要请求数据库资源，那么就会请求mysql服务器获取保存的资源，然后组织业务信息，最后按照给定的逻辑通过网络在发送给用户的浏览器。浏览器接收到返回数据后呈现给用户观看。大致的过程如下：

1. 客户发送资源请求，整个过程遵从HTTP协议
2. 然后底层协议栈通过路由算法找到合适的路由，最终发送给服务器
3. 服务器一层层解析发送过来的协议数据，最终按照应用层的HTTP协议获取用户请求信息
4. 如果请求的是静态页面nginx服务器会直接通过协议把数据发送给之前请求的客户
5. 如果有动态页面则调用php解释器解释执行，这之间包括读取缓存或者数据库
6. 在解释器执行完成之后会把数据交付给服务器由服务器通过协议把数据返回给客户呈现，正如图2.1 拓扑结构所示。

Master

Load configuration

Launch workers

Non-stop upgrade

Worker

ht\_core

ht\_upstream

ht\_proxy

ht\_fastcgi

Worker

ht\_core

ht\_upstream

ht\_proxy

ht\_fastcgi

Worker

ht\_core

ht\_upstream

ht\_proxy

ht\_fastcgi

HTTP

Web server

HTTP/HTTPS

FastCGI

Application server

memcache

Memcached

Advanced I/O

sendfile,AIO

mmap etc.

Cache loader

Cache manager

Backend

Multiplexing

via kevent/

epoll/select

proxy

cache

图2.1 PHP应用工作原理

* + 1. 选择PHP的优势
  1. 代码开源：这样就能得到所有的PHP实现，对于某些特殊的需求可以实现按需修改
  2. 免费性：任何商业公司，成本是一个不得不考虑的问题。和其他技术相比PHP不仅高效稳定而且是免费的。选择它作为技术实现没有没有成本费用。
  3. 快捷性：相对其他语言php学习成本低，程序开发时间短，运行也比较快。编程简单高效。语言本身提供很多内置高效的函数库来处理一些业务逻辑。
  4. 跨平台：PHP是服务端的脚本，可以运行在UNIX、LINUX、WINDOWS、Mac OS、ANDROID等主流系统上。
  5. 图像处理：可以使用PHP动态创建图像，PHP图像本身默认使用GD2来做图像处理，当然也可以通过配置使用image magick进行图像处理。
  6. 支持面向对象：在我们项目选择的5.0以后的版本中，PHP 能够很好的支持一些面向对象特性编程。这样就能对于我们开发大型复杂的系统提供支持。同时应对复杂多变的促销需求，使用面向对象技术可以使用敏捷开发。
  7. 有很多成熟的框架，比如ThinkPHP, CodeIgnode, laveral等基于MVC的框架。能够很好应对开发需求。
  8. MySQL 简介

MySQL由于其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，许多中小型网站为了降低网站总体拥有成本而选择了MySQL作为网站数据库。其主要特点：

* + 1. 使用C和C++编写，并使用了多种编译器进行测试，保证源代码的可移植性
    2. 支持FreeBSD、Linux、Mac OS、Windows等多种主流操作系统
    3. 为多种编程语言提供了API。这些编程语言包括C、C++、Python、Java、Perl、PHP、Eiffel、Ruby和Tcl等。
    4. 支持多线程，充分利用CPU资源
    5. 优化的SQL查询算法，有效地提高查询速度
    6. 既能够作为一个单独的应用程序应用在客户端服务器网络环境中，也能够作为一个库而嵌入到其他的软件中提供多语言支持，常见的编码如中文的GB 2312、BIG5等
    7. 提供TCP/IP、ODBC和JDBC等多种数据库连接途径
    8. 提供用于管理、检查、优化数据库操作的管理工具
    9. 可以处理拥有上千万条记录的大型数据库
  1. 促销推荐简介
     1. 促销商品热度数据

本项目使用的热度数据，主要用于反应促销商品的一个喜爱程度。促销商品的热度主要从商品的购买数量，商品的关注数量，商品的销售时间属性，热度数据的生成主要是利用Hadoop技术从不同的维度去统计。相关Hadoop的介绍参考2.6小结。

* + 1. 促销系统推荐系统

推荐系统和检索系统都是为了解决用户信息过载的产物。相比检索系统推荐系统主要解决的是挖掘长尾用户，检索系统而言一般而言用户知道要什么所以说用户有明确的关键字。而作为电子商务促销系统平台，促销活动的发布是有企业运营人员发布的。用户不知道有什么样子的促销活动。但是用的注册信息，用户的购买行为明确的指示我们他们的喜好。本项目主要采用基于用的协同过滤和基于物品的协同过滤算法混合算法。

协同过滤算法正如其名字，是大家一起协同把那些不相关的过滤掉。由此引出基于物品的协同过滤算法和基于用户的协同过滤算法。正如其名基于用户的协同过滤算法是看看你买了什么，我拿你这个人的购买列表参考下。而基于物品的协同过滤算法是买了这个商品的人一般还会买别的相关商品。比如买了数据挖掘书籍的很很可能再去买机器学习。这两种算法各有优缺点。总的来说都是体现物以类聚的思想。因原理类似以基于用户的协同过滤算法进行深入研究。

基于用户的协同过滤算法，在本项目中是应用场景比较多，因为不同地域，不同餐饮部门在购买商品的时候往往有一定的相关性。比如湘菜菜系的餐馆，购物车就有很大的相似性。如何定义两个用户的相似性，电子商务网站一般使用购物车来分析相关的相似性。以用户A及用户B的购物车类比定义。相似度大致可以定义为：

相似度 ＝ |A ∩ B| / | A ∪ B|

或者通过余弦相似度去计算：

余弦相似度 ＝ |A ∩ B| /

大致的意思就是AB两个用户他们购物车相同的购买商品的个数除以他们所有的商品个数。既然是要推荐，我们计算出相似度以后，还是远远不够的，一个用户对一个商品感兴趣的程度，是通过与他兴趣相似的用户购买的记录中推荐出来的。可以用如下的公式表示一个用户对一个商品的感兴趣程度：

如何评价一个好的推荐系统，业界一般使用如下一些指标：

* 1. 预测准确度
  2. 覆盖率：对于商品长尾理论的发掘能力，让尽量少的商品被淹没
  3. 多样性：用户背景环境多样，使用场景的多样，多样性主要用来描述能够覆盖多少不同类型的用户
  4. 新颖性：用户可能不希望推荐它一些他一定会去买的商品，因为这样的推荐可能等于没有推荐。能否推荐一些他没有买过的商品。比如说：“瓶酒炸鸡”当一中东西形成潮流，某个用户的相似用户可能会响应潮流，相似用户的购物车的变化会被推荐到和这些用户有相似行为的用户等。这样的推荐系统才能给用户带来新颖和惊喜
  5. 促销系统运行流程

在该电商公司，整个业务线分为前端和后端业务线。在这里要说明的这里的前端和后端划分跟我们web技术的前后端不是一个概念。在这里业务的划分是根据用户购买的关系划分的。前端业务主要负责商品的销售，而后端主要负责商品的库存和物流仓库部分。促销系统是为了营销活动而衍生出的一个依赖于商品管理系统，商品销售管理系统，订单系统等。从一个促销活动的发布到用户购买商品享受一个促销优惠的活动大致的过程为：首先运营人员根据相关的数据，分析策划要发布促销活动，然后就在促销活动的录入接口填写相关的活动规则，活动一发布默认不会上线也就不会设置为有效。而是等运营人员把促销活动设置为上线状态后，在商城普通的用户才能看到相关的促销活动，相关的促销活动可能有很多。用户看到的相关活动时根据用户的购买记录利用算法计算出相关促销商品的热度，然后调用促销推荐系统计算出每个用户可能感兴趣的商品把他保存到数据库中。这样当用户登录的时候就能快速的查看到自己可能感兴趣的促销活动。如果用点击了该商品加入到购物车，购物车在结算的时候就会根据用户的信息已经购买的商品信息找出符合条件的促销活动信息，根据这些信息去创建订单，然后把订单中对每个商品的优惠信息都保存到订单系统中这样方便以后的售后服务比如说退货应该给用户退换多少金额等至此一个促销系统的运行过程才基本完成。正如上文所诉，促销系统整个业务流程可以使用图2.2所示。

用户浏览商品信息

显示相关促销商品

运营

发布活动

规则检测

是否合法

禁止发布促销活动

创建活动

存储活动

上线活动

促销热度分析

计算价格

均摊促销

创建订单

是否有优惠券

购物流程结束

发送优惠券给用户

进入物流系统

否

是

图2.2 促销系统运行示意图

* 1. Hadoop 简介

Hadoop是一个由Apache基金会所开发的分布式系统基础架构。用户可以在不了解分布式底层细节的情况下，开发分布式程序。充分利用集群的威力进行高速运算和存储。任何东西都有两面性正如hadoop也是一样。

1. Hadoop集群的扩展性是其一大特点，Hadoop可以扩展至数千个节点，对数据持续增长，数据量特别巨大的需求很合适。
2. Hadoop的成本是其另一大优势，由于Hadoop是开源项目，而且不仅从软件上节约成本，硬件上的要求也不高。目前去IOE潮流风行，低成本的Hadoop也是一大推手。
3. Hadoop生态群活跃，其周边开源项目丰富，HBase, Hive，Impala等等基础开源项目众多。

同时hadoop的缺点也是很明显的：

* 1. 全量场景，任务内串行
  2. 重吞吐量，响应时间完全没有保证
  3. 中间结果不可见，不可分享
  4. 单输入单输出，链式浪费严重
  5. 链式MR不能并行
  6. 粗粒度容错，可能会造成陷阱
  7. 图计算不友好
  8. 迭代计算不友好

但是在我们的促销推荐系统，因为用户的购物活动不是那么的频繁一天一般来说就一次。也就是当天购买当天所需的商品即可。所以说在我们促销推荐系统中对hadoop的时间要求不是很高。同时我们关系的只是通过推荐算法计算出来的，相关协同过滤算法的结果而已。即哪些促销活动用户可能更感兴趣。同时公司本身也有hadoop系统集群。可以0部署安装使用。我们要做的就是简单的根据算法在空闲时间比如说凌晨在服务器压力比较小的时候编写定时脚本计算用户可能喜欢的活动然后保存结果到数据库，然后用户在登录的时候就可以直接查询出相关的结果。

* 1. 本章小结

本章就促销系统所使用的各项技术做了系统的介绍。首先介绍了WEB服务器的选择，分析论证了nginx和apache各自的优缺点。并且结合业务场景分析了选择nginx的原因。同里分析了PHP和MySQL数据库。最后介绍了为了环境用户对促销活动信息过载或者说不对称而引入的促销推荐系统，旨在让用户最快的获取自己感兴趣的促销，这部分主要论证了为什么选择推荐系统的基于用户和物品的混合协调过滤算法。并在这个基础上描述了到了使用的相关分布式计算。

# 系统需求分析和概要设计

* 1. 系统需求分析

经过与产品经理的多次沟通，整理出来关于本系统关于促销系统以下一些详细的需求。包括促销对象范围的确定，促销活动范围的确定和具体的促销活动相关内容。本促销系统的需求大致如下：

* + 1. 功能性需求
  1. 能够使用DM标签系统快速的给指定的用户发放促销活动
  2. 促销活动能够制定时间周期
  3. 促销活动的范围能够指定到具体的分类或者商品，如可以指定只对蔬菜（分类）进行促销。或者更具体的指定如只对哪个商品进行促销
  4. 要能指定某个渠道的用户参加促销活动
  5. 要能根据活动的最晚配送时间，只要满足最晚配送时间的才能够参加活动
  6. 用户购买的商品如果发生退货要能正确的计算出参与活动的哪些商品退货了，每个商品优惠了多少。用户退货能够计算出退货价格。
  7. 要支持兼容之前的微信立减5元促销活动和之前的满减活动。之前只有这两种活动
  8. 要能指定促销活动的支付类型，只要支付方式满足的用户才能参与促销。比如发布一个促销活动只要微信支付的人才能参与促销，那么如果用户选择了货到付款就不能参与促销活动。
  9. 因为促销系统很多都是低成本运行，有些甚至亏损运行，所有必须要有限购功能。限购的功能有两个维度，对于每个人的限购和所有的商品的限购。或者其中的任意一中组合。其中如果限购都不输入则代表没有限购。
     1. 非功能性需求
  10. 系统要求稳定；
  11. 系统要求每秒并发访问500以上的访问量；
  12. 提供相关的接口文档；
  13. 系统的响应速度不能低于2s；
  14. 系统出错能够快速的重启恢复。
  15. 概要设计

本小结首先介绍了跟促销系统相关的子系统的构造结构和相互关系。并通过分析一个促销系统的发布，到用户购买商品享受促销活动，再到最后的退货等售后服务。促销系统相关的模块众多。首先，促销系统促销的商品要有库存这关联到wms（仓库管理系统）。然后是必需要有商品信息,这设计到pms(商品管理系统)，同样的物品在不同的地区售价和规格也不相同，由物品变为商品要在sms（商品售卖系统里面）设置。然后用户在购物的时候才能看到有促销活动，用户在下单购物的时候要创建订单这设计到订单系统，然后物流系统根据订单去配送商品。所以一般来说跟促销系统相关的电商子系统包括商品管理系统，仓库管理系统，销售管理系统，和订单系统。如下图所示：

图3-1 促销系统相关模块结构

* + 1. 商品管理系统

商品管理系统，主要的职责就是把采购过来的产品进行加工处理变成预卖商品。在库存中的形式就是形成一个SKU=Stock Keeping Unit（库存量单位）

主要的功能就是显示相关货物信息，添加货物信息，修改货物信息，删除活动信息，添加货物分类映射。其中添加一个货物信息一般主要指货物的规格属性包括：名称、所属分类信息、规格、图片、保质期、重量、库存等货物基本属性。在这里添加的货物跟商品的最大的区别就是没有价格信息。同时一样东西在不同的地区，人们对他有不同的叫法，为了能够让不同地域的用户在浏览商城的时候能够清晰的知道商品的名称做了一个这样的分类映射，方便地域性显示。商品管理系统里是不售卖的商品，促销系统跟商品管理系统有关系是因为有满赠类的活动，满赠的赠送物品可以是售卖的也可以是不售卖的，但是赠送给用户肯定是必须存在的物品。整个商品管理系统结如图3-2所示。

图3-2 商品管理系统结构图

* + 1. 商品销售管理系统

商品销售管理系统主要集中跟销售运营的所有活动。可以简单的理解为在物品上添加价格。以及发布相关运营活动的这样一个系统。所以整个销售管理系统大致包括的功能就是商品管理，商品广告管理，优惠券管理等。其中商品管理包括商品列表显示，商品发布，商品价格的制定；广告位管理发布广告商品信息，广告商品信息列表，下架广告商品等；优惠券包括发布优惠券活动的规则，发布优惠券活动，系统中优惠券设计两种方式一种是用户促销的优惠活动，另外一种是系统促发的优惠券活动，商品销售管理系统是整个电商平台的一个重要组成部分。该系统集成了所有的跟商品运营活动有关的功能，促销系统最终也要集成到销售管理系统上给运营人员发布促销活动。整个商品销售管理系统的拓扑图如图3-3。

图3-3 商品销售管理系统结构图

* + 1. 订单系统

任何订单流程都是电子商务的核心。处于商业安全的原因，这里只大概的讲述下跟促销系统相关的一些订单创建的流程。用户在浏览电商网站购买商品的时候，相关的商品列表，商品详情只会显示那些商品销售管理子系统中发布的商品。所有的商品都会去读取商品信息表。商品在商品销售系统中发布商品的时候会创建一条商品。一般来说会保护该商品的一些基本属性和价格属性。然后再结合促销系统，就可以确定哪类商品或者哪个商品有促销活动。促销活动有相关促销活动关键字和促销广告语。会在之后的促销相关设计或者表结构设计中有体现。

然后用户再浏览商品的时候就能看到有促销信息。如果用户把商品添加进入购物车，那么商品会调用促销活动相关接口检查购物车商品是否满足相关的促销活动规则。比如说会查看购物车购买商品的价格，用户选择的支付方式，用户所在的地域信息，用户购物的时间，用户的历史购买记录等去刷选出符合用户的促销活动信息。如果满足，用户就会把相关的促销活动id号一同提交给订单系统。然后在订单系统里面，我们要计算是哪些商品满足促销活动，哪些商品不满足促销活动。比如说满减活动，那么我们要把相关的优惠信息均摊到每个商品最小单位。然后持久化到订单系统里面。这样我就能根据订单的情况收取相关费用和根据订单系统记录的数据提供售后服务。比如说退货，设计到促销活动的应该如何给用户退钱。这也是整个促销系统的实现设计到比较核心的一个流程。具体的流程如下图4-3所示。

图4-3 促销活动使用流程示意图

促销活动整体上可以分成两大部分，即运营发布促销活动，用户使用促销活动（包括促销推荐）。整个流程为用户登录商城看到促销推荐系统推荐好的促销活动，用户可能买也可能不买。用户看到自己想要购买的商品加入购物车，促销活动有很多类型：比如说单品促销，套装促销，满赠，满减等。用户把一些参与促销活动的商品加入到购物车，系统通过分析购物车中的商品，提示用户加买什么商品可以享受什么优惠。或者还差多少钱能够参与什么类型的活动。等用户提交了订单，用户可能对我们提供的服务、商品不满意要提出退货处理。但是用户购买的商品都是参加了相关的优惠活动的。因此所有的订单系统，在创建订单的时候要明确的记录下哪些商品参加了什么类型活动。另外由于物流的原因有些商品可能不能一并交付给客户。需要按照一定的规则把订单拆分成相关子订单。然后优惠的信息统计到订单和子订单上。并记录所购买的商品哪个是参与了促销活动的，而哪个没有。这样的设计就可以保证用户在提出退货的时候能够清晰的知道应该怎样给用户返现。整个促销系统设计的流程如图3-4。

图3-4 促销活动订单状态图

* 1. 促销系统

本章前文分别从总体和局部说明了促销系统在整个电商的应用部署环境，然后再分别介绍跟促销系统相关的子系统的功能和架构特点。在接下来的这个小结中将详细讨论如何设计促销系统，并完美的整合到该电商公司的所有相关系统中去。

促销系统的本质就是要能够发布促销活动，提高运营效果。但是公司发布促销活动，往往是做出了让利运营的，有些可能亏损运营。所以如何控制多个促销活动之间不发生冲突。是促销系统设计的一个重点。同时促销活动种类繁多，如何设计一个良好的框架，能够对未来扩展促销活动类型做好铺垫。

* + 1. 促销活动规则

促销活动规则主要的作用就是规定当有多个促销活动同时满足的时候以一种怎样的规则就选择多个同时满足的促销活动中的一个活动。在本项目当前的需求是要支持五种促销活动，分别为单品促销，套装促销，满赠，满减，打折。单品促销活动就是只针对某个特定的商品做促销活动。同时运营也可能需要发布对某一批商品进行促销活动。所以除了单品促销和套装促销活动是制定一个或几个特定的商品集合。其他的促销活动都是可以有两个维度来发布活动的。即在发布活动的时候除了可以指定某个特定的商品参加促销活动，也可以指定某些分类参加活动。没中类型都有两种可能所以总的来说发布促销活动有4种可能的范围如表3-1。

表3-1 促销活动规则范围

空－空，空－非空，非空－空，非空－非空

怎样保证让同一个商品不会出现在两个促销活动中。一种可行的办法就是设置优先级。但是如果把优先级交由运营人员在不同的人，或者同一个运营人员在不同的时间段内可能会把优先级设为相同而退化为没有优先级。如果程序给固定死优先级，这样必然带来添加新促销，或者改动促销优先级带来的开发成本。同时软件敏捷性不高。比较折中的办法是配置优先级，在给定的优先级下，如果一个商品满足多个促销活动有限按照优先级高的处理。

但是规则这样的优先级还不够，假如给蔬菜做满减活动满200减10元，同时对某个蔬菜的子分类比如说叶菜做促销满赠，比如说买五赠一。这样就导致了有些促销活动不能叠加。有时候为了增大促销力度，可以有叠加活动的促销。

* + 1. 促销活动创建

活动的创建主要就是在给定的规则下要让活动的创建有足够的灵活性。对新的促销活动能够快速的支持而不要对以前的代码进行修改实现开闭原则。活动的创建流程可以理解为首先验证运营表单数据的正确性，然后依据促销规则信息进行检查，如果都合法就创建促销活动。促销活动的创建流程如图3-5。

图3-5 促销活动创建流程图

* + 1. 促销活动均摊逻辑设计
    2. 促销活动热度数据
    3. 促销活动推荐
  1. 本章小结

本章通过对需求的深入分析，并且结合现有系统的特点。分别介绍了和促销系统相关的商品管理系统和销售管理系统已经订单系统的特点。在了解这些系统的特点和新促销系统的需求分析，结合系统论述了在结合相关系统的情况下，分析了促销系统的开发和大致实现。下一章中将对这些相关模块做详细的说明。

# 系统详细设计

本章主要是在之前章节的概要设计的基础上，详细设计促销系统各功能的实现细节。

* 1. 商品发布模块
  2. 订单系统模块
  3. 促销系统模块

1. 3. 1. 促销系统发布模块
      2. 促销系统推荐模块
   4. 本章小结

# 数据库设计

* 1. 数据模型
  2. 数据模型详细描述

整个促销系统设计的数据库表众多，主要的一些表结构下面一些表格所示：促销活动表，主要包含促销活动id,促销活动的广告语，促销活动的开始和结束时间，促销活动指定的用户范围，促销活动的规则属性等，主要的表字段信息如表5.2所示。

表5.2 促销活动表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 长度(字节) | 可否为空 | 备注 |
| 1 | id | int | 4 | 否 | 促销活动ID |
| 2 | Ads\_words | text | 0 | 是 | 广告语 |
| 3 | Promotion\_type | tinyint | 1 | 否 | 促销类型 |
| 4 | Rule\_ids | varchar | 200 | 否 | 活动规则id |
| 5 | source | int | 4 | 否 | 来源 |
| 6 | Location\_id | int | 4 | 否 | 促销地区 |
| 7 | User\_ids | text | 0 | 是 | 可参与用户 |
| 8 | Start\_time | Int | 4 | 否 | 开始时间 |
| 9 | End\_time | int | 4 | 否 | 结束时间 |
| 10 | Last\_deliver\_time | int | 4 | 是 | 最晚配送时间 |
| 11 | Pay\_type | Tinyint | 1 | 否 | 支付类型 |
| 12 | Status | Tinyint | 1 | 否 | 活动状态 |

单品促销规则表主要保存的对某个单品进行促销活动的一些指定规则信息比如说对什么商品进行促销，促销的价格，促销商品对每个用户的限购信息，整体的限购信息等。

表5.3 单品促销活动规则表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 长度（字节） | 可否为空 | 备注 |
| 1 | Id | int | 4 | 否 | 单品促销规则id |
| 2 | Sku\_numbers | int | 4 | 否 | 指定促销商品 |
| 3 | Limit | Int | 4 | 是 | 每个用户限购数量 |
| 4 | Total\_limit | int | 4 | 是 | 整体限购数量 |

满减促销活动规则表主要包含活动规则范围,同满赠活动规则一样。范围包含两个维度一个为指定SKU(商品最小单元),另外一种为指定一个分类。

同时包含指定的规则信息，规则信息支持阶梯活动，比如满100是怎样，满200怎样。同时和单品促销一样也用limit表示个人限购，total\_limit来用总体限购。

表5.4 满减促销活动规则表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 长度（字节） | 可否为空 | 备注 |
| 1 | Id | int | 11 | 否 | 满减活动规则id |
| 2 | Sku\_numbers | Text | 0 | 是 | 商品范围，如果为空代表不限制 |
| 3 | Category\_ids | Text | 0 | 是 | 分类范围，空不限制 |
| 4 | rules | Text | 0 | 否 | 活动规则 |
| 5 | Limit | Text | 0 | 是 | 个人限购 |
| 6 | Total\_limit | Text | 0 | 是 | 总体限购 |

满赠活动活动表与满减活动表基本类型，除了多了一个规则类型来区分满赠活动赠送的是商品还是优惠券。

表5.5 满赠促销活动规则表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 长度（字节） | 可否为空 | 备注 |
| 1 | Id | int | 4 | 否 | 满赠活动ID |
| 2 | Sku\_numbers | Text | 0 | 是 | 活动商品范围 |
| 3 | Category\_ids | Text | 0 | 是 | 活动分类范围 |
| 4 | Rules\_type | tinyint | 1 | 否 | 赠送类型 |
| 4 | rules | Text | 0 | 否 | 活动规则 |
| 5 | limit | Text | 0 | 是 | 个人限购 |
| 6 | Total\_limit | Text | 0 | 是 | 总体限购 |

订单表是任何一个电商的核心，设计的表字段和信息也是非常多的。因本项目只是设计到促销相关的信息，所以只列出和促销有相关的字段信息如表5.6所示。

表5.6 商品订单表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 长度（字节） | 可否为空 | 备注 |
| 1 | id | Int | 4 | 否 | 订单ID |
| 2 | Order\_number | Char | 32 | 否 | 订单号 |
| 3 | username | char | 32 | 否 |  |
| 4 | User\_id | int | 4 | 否 |  |
| 5 | Total\_price | int | 4 | 否 |  |
| 6 | Deal\_price | int | 4 | 否 |  |
| 7 | City\_id | int | 4 | 否 |  |
| 8 | Deliver\_time | int | 4 | 否 |  |
| 9 | Deliver\_date | int | 4 | 否 |  |
| 10 | Line\_id | int | 4 | 否 |  |
| 11 | Minus\_amount | int | 4 | 否 |  |
| 12 | Pay\_type | int | 4 | 否 |  |
| 13 | Promotion\_ids | varchar | 1024 | 是 | 订单促销id |

因业务需求问题，有些商品不能一起送达，当用户下单后，系统会根据相关规则把一个订单拆分可以配送的子弹。配送按照子单进行。一个母单对应多个多个子单。和促销系统相关的字段信息如表5.7所示。

表5.7 商品子订单列表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 长度（字节） | 可否为空 | 备注 |
| 1 | id | int | 4 | 否 | 子单id |
| 2 | Order\_number | char | 32 | 否 | 订单编号 |
| 3 | Total\_price | int | 4 |  |  |
| 4 | Deal\_price | int | 4 |  |  |
| 5 | Location\_id | Int | 4 |  |  |
| 6 | Minus | Int | 4 |  |  |
| 7 | Order\_id | int | 4 |  |  |
| 8 | Order\_resource | Tinyint | 1 |  |  |
| 9 | Promotion\_ids | varchar | 1024 |  |  |

除了记录订单和子订单的信息我们还是没有办处理相关售后服务每天限购和整体限购的逻辑，详情见表5.8。

表5.8 商品订单详情表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 数据类型 | 长度（字节） | 可否为空 | 备注 |
| 1 | id | int | 4 |  | 订单详情id |
| 2 | Order\_id | Int | 4 |  |  |
| 3 | quantity | Int | 4 |  |  |
| 4 | Price | Int | 4 |  |  |
| 5 | Sku\_number | Int |  |  |  |
| 6 | Category\_id | Int | 4 |  |  |
| 7 | Single\_price | int | 4 |  |  |
| 8 | Suborder\_id | int | 4 |  |  |
| 9 | Promotion\_ids | varchar | 1024 |  |  |
| 10 | Promotion\_quantity | int | 4 |  |  |

* 1. 本章小结

本章主要针对促销系统相关的数据库表设计。从数据库层面上展示了整个促销系统各个数据关系之间的相互作用，并且对相关重要的表结构做了详细的介绍。在下一章的系统实现中将详细的论述如何使用到这些表数据信息。

# 系统的实现

* 1. 发布促销活动
  2. 促销显示
  3. 促销结算
  4. 促销商品退货计算
  5. 促销推荐
  6. 本章小结

# 总结与展望

* 1. 总结

本论文分析了目前快速发展的互联网现状，对本项目的使用到的相关技术做了大致的介绍。并根据工程实际背景和项目需求，介绍了相关系统的设计。并根据某电商公司的需求设计和开发了这套促销系统。

本文通过概述系统设计的背景，分析各模块的相互依赖关系。然后重点分析促销系统在功能性需求和非功能性需求方面未来可能应对的挑战。使用了设计模式等相关敏捷开发的思想，不仅能够快速的拓展信息促销活动，发布新的促销规则。同时引入了促销推荐系统，能够尽可能的让用户发现可能感兴趣的促销活动。最后通过测试，很好了完成了系统需求。通过公司BI部门的统计，该促销活动上线以来，销售额不断攀升。提高了公司的业绩。

因时间和精力有限，本系统可能还有些不足：

* 1. 后台发布页面不是很美观。因为不是设计出身，很多操作界面比较生硬。
  2. 系统在监测冲突那块进行了大量的查询操作，虽然这块时面对运营人员，但是还是可以优化相关数据设计，或者使用缓存技术来改进运营发布活动的体验。因时间关系以及重点放在普通用户上，以及开发周期短，这块后期可以改进下
  3. 对于促销推荐系统，在现有的协同过滤算法上，通过统计分析商品的消耗时间，细化现有的时间维度。同时因为地域维度，可能会得到更好的推荐效果。提高促销系统的使用效率。

因此在接下的时间里面主要的工作是：因为页面美工人员美化操作窗口。通过DM标签接入用户的地域信息，通过统计用户购买没中商品的一个时间间隔，来统计一般商品的消耗时间。因为不同的用户消耗的周期不一样。所以这个统计维度主要是对同一个用户的平均值统计。然后再引入这两个变量到推荐系统里面优化推荐效果。同时添加完善目前的日志系统，提高系统的稳定性和问题的快速定位及服务恢复。

* 1. 展望

电子商务因借助现代互联网技术不仅成本低，效率高，发布消息快。必将代表未来贸易的发展方向。同时，我们当先正处在一个信息爆炸的社会，换言之就每个个体我们都处在信息过载的时代，传统的促销方式已经没有能力去解决这一个问题。利用互联网技术，利用时下的大数据技术。我们就能分析出用户需要什么，用户对什么感兴趣。然后智能的推荐相关商品给他。同时因现在互联网一个站点往往功能众多。有些需要的功能隐藏很深，不需要的功能或者商品又放在一个明显的位置，浪费空间。未来甚至可以利用大数据技术深度挖掘用户行为爱好，为每个人生成一个深度定制化的电商网站。提供的服务类似与每位顾客的私人助理。同时相对而言，大数据的思维在信息技术的今天实现也相对较为容易，为生活在这样的技术时代而兴奋。相信随着规模的逐步扩大这样的设想，为期不久。

# 参考文献

［1］房爱莲. PHP动态网页设计与制作案例教程[M]. 北京：北京大学出版社，2011

# 附录A 附录示例

# 致谢

本论文是在吴中海、张齐勋老师的悉心指导下完成的。张齐勋老师作为一名优秀的、经验丰富的教师，具有丰富的专业知识和项目经验，在整个论文实验和论文写作过程中，对我进行了耐心的指导和帮助，提出严格要求，引导我不断开阔思路，为我答疑解惑，鼓励我大胆创新，使我在这一段宝贵的时光中，既增长了知识、开阔了视野、锻炼了心态，又培养了良好的实验习惯和科研精神。在此，我向我的指导老师表示最诚挚的谢意！

# 北京大学学位论文原创性声明和使用授权说明

**原创性声明**

本人郑重声明：所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文不含任何其他个人或集体已经发表或撰写过的作品或成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律结果由本人承担。

论文作者签名： 日期： 年 月 日

**学位论文使用授权说明**

（必须装订在提交学校图书馆的印刷本）

本人完全了解北京大学关于收集、保存、使用学位论文的规定，即：

* 按照学校要求提交学位论文的印刷本和电子版本；
* 学校有权保存学位论文的印刷本和电子版，并提供目录检索与阅览服务，在校园网上提供服务；
* 学校可以采用影印、缩印、数字化或其它复制手段保存论文；
* 因某种特殊原因需要延迟发布学位论文电子版，授权学校□一年/□两年/□三年以后，在校园网上全文发布。

（保密论文在解密后遵守此规定）

论文作者签名： 导师签名：

日期： 年 月 日