
网上商城购物系统毕业设计

目 录

前 言.....	1
第 1 章 绪论.....	2
1.1 课题背景.....	2
1.2 课题研究的意义.....	2
1.3 课题研究的方法.....	2
第 2 章 程序开发的技术基础.....	3
2.1 HTML 语言.....	3
2.2 JSP.....	4
2.3 SSH.....	4
2.3.1 Hibernate 的工作原理.....	4
2.3.2 Spring 的工作原理.....	5
2.3.3 Struts2 的工作原理.....	6
2.4 数据库原理.....	8
2.4.1 SQL 语言简单介绍.....	8
2.4.2 ADO 技术介绍.....	8
第 3 章 需求分析.....	10
3.1 系统主要工作流程图.....	10
3.1.1 后台管理模块流程图.....	10
3.1.2 前台管理模块流程图.....	10
3.2 系统实现采用环境.....	11
3.3 系统功能需求.....	12
第 4 章 功能的设计与实现.....	13
4.1 系统主要设计.....	13
4.1.1 系统主要功能.....	13
4.1.2 主要功能模块.....	13
4.2 数据库的分析与设计.....	14

4.2.1 数据库的需求分析.....	14
4.2.2 数据库的结构设计.....	15
4.3 系统主要功能实现.....	18
4.3.1 注册页面的实现.....	18
4.3.2 登录界面的实现.....	19
4.3.3 商品分类.....	22
4.3.4 后台管理登录页面.....	26
第5章 系统测试.....	32
5.1 测试方法.....	32
5.2 预期测试结果.....	32
5.3 实际测试结果.....	32
结 论.....	33
谢 辞.....	34
参 考 文 献.....	35

前 言

近年来，随着 Internet 的迅速崛起，互联网已日益成为收集提供信息的最佳渠道并逐步进入传统的流通领域。于是电子商务开始流行起来，越来越多的商家在网上建立起商店，向消费者展示出一种新颖的购物理念。

网上购物系统作为 B2B, B2C, C2C 电子商务的前端商务平台，在其商务活动全过程中起着举足轻重的作用。本文旨在讨论如何建设 B2C 的网上购物系统。

网上购物是一种具有交互功能的商业信息系统。它向用户提供静态和动态两类信息资源。所谓静态信息是指那些比经常变动或更新的资源，如公司简介、管理规和公司制度等等；动态信息是指随时变化的信息，如商品报价，会议安排和培训信息等。网上购物系统具有强大的交互功能，可使商家和用户方便的传递信息，完成电子贸易或 EDI 交易。这种全新的交易方式实现了公司间文档与资金的无纸化交换。

在美、日等信息化程度较高的国家和地区，网络商店发展速度迅猛，美国的世界级超一流的零售商，如沃尔玛凯玛特，家庭仓储、科罗格、J.C 培尼等纷纷跻身于网络经商的行列。调查表明：美国的家庭已越来越习惯于在家中从网上购物。

在我国，网上购物从无到有也只不过短短几年时间。我国第一家网上购物发生在 1996 年，燕莎友谊商场首次通过网上商城售出一个景泰蓝，虽然货款的支付不是在网上进行的，但这毕竟为我国零售业奏出了网上购物的先声。根据 CNNIC 的统计结果，截止 2004 年 1 月中国有互联网用户 7950 万人，而又有 40.7% 以上的网民在过去的一年里有过网上购物经历，这就说明中国有网上购物的用户达是 3235.7 万人，并且这一数字还在以 17% 左右的速度增长，这说明在中国发展网上购物具有良好的群众基础，网上购物方式日趋被大家所接受。

第 1 章 绪论

1.1 课题背景

随着 Internet 的快速发展,人们已经进入了信息时代,Web 已经对商业、工业、银行、财政、教育、政府、娱乐和信息产业及人们的工作、生活产生了深远的影响。许多传统的信息和数据库系统正在被移植到互联网上,人们获得信息的来源已经不再只是报纸、电视、广播等传统媒体,一个新的信息传送媒体逐渐被人们所青睐。围广泛的,复杂的分布式应用正在 Web 环境中出现。网上购物系统就相继的出现,给人们带来了不少的方便快捷,网上购物系统正在快速向前发展。

1.2 课题研究的意义

当今时代是一个互联网络高速发展的时代,本课题的研究意义就在于构造一个网络购物的平台,使人们在纷繁的网络中也能够找到一个购物的天地。同时也是对三年学习的一个总结和提高。

1.3 课题研究的方法

本课题的开发采用 SSH+SQL Sever 实现,前期主要运用所学相关知识、查询资料、学习其他优秀管理系统开发方法,结合相关要求写出需求分析;后期,进行本课题的设计与开发。课题的页面主要采用 java 语言对数据库进行操作,通过对数据库的查询搜索来实现,对表单以及数据的校验采用 java script 脚本语言,控制页面的正确显示和对用户类型的权限设计。采用 SQL 数据库,实现整个系统的操作。在整个设计过程中以需求分析为基础,写出系统实现流程及相关问题的实现方法;系统开发完成后,进行调试和运行,做好调试和试运行的相关记录。

第 2 章 程序开发的技术基础

2.1 HTML 语言

HTML (Hyper Text Markup Language 超文本标记语言) 是一种用来制作超文本文档的简单标记语言。与常见的字处理文件不同, web 页以超文本标识语言编排格式。HTML 文件是带有特定 HTML 插入标记的用以编排文档属性和格式的标准文本文件。它能独立于各种操作系统平台 (如 UNIX, WINDOWS 等)。自 1990 年以来 HTML 就一直被用作 World Wide Web 上的信息表示语言, 用于描述 Homepage 的格式设计和它与 WWW 上其它 Homepage 的连结信息。

HTML 文档 (即 Homepage 的源文件) 是一个放置了标记的 ASCII 文本文件, 通常它带有 .html 或 .htm 的文件扩展名。生成一个 HTML 文档主要有以下三种途径:

(1) 手工直接编写 (例如用你所喜爱的 ASCII 文本编辑器或其它 HTML 的编辑工具)。

(2) 通过某些格式转换工具将现有的其它格式文档 (如 WORD 文档) 转换成 HTML 文档。

(3) 由 Web 服务器 (或称 HTTP 服务器) 一方实时动态地生成。
一般的 HTML 页面具有如下的结构:

```
<html>
  <head>
    <title>,<base>,<link>,<is in
dex>,<meta>
  </head>
  <body>
```

这 是

HTML 正文部分

<

/body>

</html>

可见在一个 HTML 网页文件中，一般必须有一对 <html> </html> 标记作为文件的开头和结尾，在 <html> 标记后是头部标记 <head></head>，其后是实体标记 <body></body>。

2.2 JSP

JSP (JavaServer Pages) 是由 Sun Microsystems 公司倡导、许多公司参与一起建立的一种动态网页技术标准。JSP 技术有点类似 ASP 技术，它是在传统的网页 HTML 文件 (*.htm, *.html) 中插入 Java 程序段 (Scriptlet) 和 JSP 标记 (tag)，从而形成 JSP 文件 (*.jsp)。

Web 服务器在遇到访问 JSP 网页的请求时，首先执行其中的程序段，然后将执行结果连同 JSP 文件中的 HTML 代码一起返回给客户。插入的 Java 程序段可以操作数据库、重新定向网页等，以实现建立动态网页所需要的功能。

JSP 与 Java Servlet 一样，是在服务器端执行的，通常返回该客户端的就是一个 HTML 文本，因此客户端只要有浏览器就能浏览。

JSP 的 1.0 规的最后版本是 1999 年 9 月推出的，12 月又推出了 1.1 规。目前较新的是 JSP1.2 规，JSP2.0 规的征求意见稿也已出台。

2.3 SSH

2.3.1 Hibernate 的工作原理

1. Hibernate 是如何连接数据库

主要是通过 hibernate.cfg.xml 配置文件中的配置。

在这个文件中定义了数据库进行连接所需要的信息，包括 JDBC 驱动、用户名、密码、数据库方言等 configuration 类借助 dom4j 的 XML 解析器解析设置环境，然后使用这些环境属性来生成 SessionFactory。这样这个 sessionFactory 生成的 session 就能成功获得数据库的连接。

2. Hibernate 是如何进行数据库写操作

对数据库的写操作包括保存、更新和删除，当保存一个 POJO 持久对象时，触发 Hibernate 的保存事件监听器进行处理。Hibernate 通过映射文件获得对象对应数据库表名以及属性所对应的表中的列名，然后通过反射机制持久化对象（实体对象）的各个属性，最终组织成向数据库插入新对象的 SQLinsert 语句。调用了 session.save() 方法后，这个对象会标识成持久化状态存放在 session 中，对于 Hibernate 来说它就是一个持久化了的对象，但这个时候 Hibernate 还不会真正的执行 insert 语句，当进行 session 的刷新同部或事务提交时，Hibernate 会把 session 缓存中的所有 SQL 语句一起执行，对于更新、删除操作也是采用类似的机制。

然后，提交事务并事务提交成功后，这些写操作就会被永久地保存进数据库中，所以，使用 session 对数据库操作还依赖于 Hibernate 事务的处理。如果设置了二级缓存，那么这些操作会被同步到二级缓存中，Hibernate 对数据库最终操作也是依赖于底层 JDBC 对数据库进行。

3. Hibernate 如何从数据库中载入对象

当使用 session.load() 载入对象时，可以设置是否采用延迟加载，如果延迟加载，那么 load 返回的对象实际是 CGLIB 或 javassist 返回的代理类，它的非主键属性都是空的，这对于对象集合属性很有效。Hibernate 以此来节约存，当真正需要读取对象时，Hibernate 会先尝试从 session 缓存中读取，如果 session 缓存中数据不存在或者是脏数据并且配置了二级缓存，Hibernate 尝试从二级缓存中检索数据，否则 Hibernate 会根据对象类型，主键等信息组织 select 语句到数据中读取，再把 select 结果组织成对象返回。

4. Hibernate 如何进行数据库查询操作

Hibernate 提供 SQLHQLCriteria 查询方式。HQL 是其中运用最广泛的查询方式。用户使用 session.createQuery() 方法以一条 HQL 语句为参数创建 Query 查询对象后，Hibernate 会使用 Anltr 库把 HQL 语句解析成 JDBC 可以识别的 SQL 语句，如果设置了查询缓存，那么执行 Query.list() 时，Hibernate 会先对查询缓存进行查询，如果查询缓存不存在，再使用 select 语句查询数据库。

2.3.2 Spring 的工作原理

Spring 是非侵入式的：典型地，Spring 应用中的对象不依赖于 Spring 的特定类。

1. 控制反转

Spring 通过一种称作控制反转 (IoC) 的技术促进了松耦合。当应用了 IoC, 一个对象依赖的其它对象会通过被动的方式传递进来, 而不是这个对象自己创建或者查找依赖对象。你可以认为 IoC 与 JNDI 相反不是对象从容器中查找依赖, 而是容器在对象初始化时不等对象请求就主动将依赖传递给它。

1. 面向切面

Spring 提供了面向切面编程的丰富支持, 允许通过分离应用的业务逻辑与系统级服务 (例如审计和事务管理) 进行聚性的开发。应用对象只实现它们应该做的——完成业务逻辑——仅此而已。它们并不负责 (甚至是意识) 其它的系统级关注点, 例如日志或事务支持。

2. 容器

Spring 包含并管理应用对象的配置和生命周期, 在这个意义上它是一种容器, 你可以配置你的每个 bean 如何被创建——基于一个可配置原型 (prototype), 你的 bean 可以创建一个单独的实例或者每次需要时都生成一个新的实例——以及它们是如何相互关联的。然而, Spring 不应该被混同于传统的重量级的 EJB 容器, 它们经常是庞大与笨重的, 难以使用。

3. 框架

Spring 可以将简单的组件配置、组合成为复杂的应用。在 Spring 中, 应用对象被声明式地组合, 典型地是在一个 XML 文件里。Spring 也提供了很多基础功能 (事务管理、持久化框架集成等等), 将应用逻辑的开发留给了你。所有 Spring 的这些特征使你能够编写更干净、更可管理、并且更易于测试的代码。它们也为 Spring 中的各种模块提供了基础支持。

2.3.3 Struts2 的工作原理

Struts2 并不是一个陌生的 web 框架, 它是以 Webwork 的设计思想为核心, 吸收 struts1 的优点, 可以说 Struts2 是 Struts1 和 Webwork 结合的产物。

Struts2 的工作原理: 一个请求在 Struts2 框架中的处理分为以下几个步骤:

- (1) 客户端发出一个指向 servlet 容器的请求 (tomcat)
- (2) 这个请求会经过几个过滤器, 最后会到达 FilterDispatcher 过滤器。

(3) 过滤器 `FilterDispatcher` 是 Struts2 框架的心脏，在处理用户请求时，它和请求一起相互配合访问 Struts2 的底层框架结构。在 web 容器启动时，Struts2 框架会自动加载配置文件里相关参数，并转换成相应的类。如：`ConfigurationManager`、`ActionMapper` 和 `ObjectFactory`。`ConfigurationManager` 存有配置文件的一些基本信息，`ActionMapper` 存有 action 的配置信息。在请求过程中所有的对象（`Action`，`Results`，`Interceptors`，等）都是通过 `ObjectFactory` 来创建的。过滤器会通过询问 `ActionMapper` 类来查找请求中需要用到的 `Action`。

(4) 如果找到需要调用的 `Action`，过滤器会把请求的处理交给 `ActionProxy`。`ActionProxy` 为 `Action` 的代理对象。`ActionProxy` 通过 `ConfigurationManager` 询问框架的配置文件，找到需要调用的 `Action` 类。

(5) `ActionProxy` 创建一个 `ActionInvocation` 的实例。`ActionInvocation` 在 `ActionProxy` 层之下，它表示了 `Action` 的执行状态，或者说它控制的 `Action` 的执行步骤。它持有 `Action` 实例和所有的 `Interceptor`。

(6) `ActionInvocation` 实例使用命名模式来调用

1. `ActionInvocation` 初始化时，根据配置，加载 `Action` 相关的所有 `Interceptor`。

2. 通过 `ActionInvocation.invoke` 方法调用 `Action` 实现时，执行 `Interceptor`。在调用 `Action` 的过程前后，涉及到相关拦截器 (`interceptor`) 的调用。

(7) 一旦 `Action` 执行完毕，`ActionInvocation` 负责根据 `struts.xml` 中的配置找到对应的返回结果。返回结果通常是（但不总是，也可能是另外的一个 `Action` 链）一个需要被表示的 JSP 或者 `FreeMarker` 的模版。在表示的过程中可以使用 Struts2 框架中继承的标签。

2.4 数据库原理

2.4.1 SQL 语言简单介绍

SQL 是英文 Structured Query Language 的缩写，意思为结构化查询语言。

SQL 语言的主要功能就是同各种数据库建立联系，进行沟通。按照 ANSI（美国国家标准协会）的规定，SQL 被作为关系型数据库管理系统的标准语言。SQL 语句可以用来执行各种各样的操作，例如更新数据库中的数据，从数据库中提取数据等。

目前大多数关系数据库管理系统，如 Oracle、Sybase、Microsoft SQL Server、Access 都采用了 SQL 语言标准。虽然很多数据库都对 SQL 语句进行了再开发和扩展，但是包括 Select、Insert、Update、Delete、Create 以及 Drop 在的标准 SQL 命令仍然可以被用来完成几乎所有的数据库操作。在众多的 SQL 命令中，select 语句应该算是使用最频繁的。select 语句主要被用来对数据库进行查询并返回符合用户查询标准的结果数据。select 语句中位于 select 关键词之后的列名用来决定哪些列将作为查询结果返回。用户可以按照自己的需要选择任意列，还可以使用通配符“*”来设定返回表格中的所有列。

select 语句中位于 from 关键词之后的表格名称用来决定将要进行查询操作的目标表格。

除了上面所提到的运算符外，LIKE 运算符在 where 条件从句中也非常重要。LIKE 运算符的功能非常强大，通过使用 LIKE 运算符可以设定只选择与用户规定格式相同的记录。

2.4.2 ADO 技术介绍

ADO (ActiveX Data Objects) 技术是微软公司推出的数据库连接技术，运用该技术可以对各种数据库（如 SQL Server、Access、Oracle 及 Sybase 等）的、查询、存取等操作。

ADO 对象给开发人员提供一种快捷、简单、高效的数据库访问方法，ADO

可以包含在脚本中来产生对数据库的连接，并从数据库表中读取数据，形成实际要使用的对象集合。而且，对于数据资源，提供了应用程序一级的界面。不过，ADO 并不与数据资源直接通信，而是通过称为 OLE DB 的中间界面，对于数据资源 (Microsoft SQL Server)，OLE DB 提供了系统一级的界面。

ADO 是一组优化的访问数据库的对象集，为 Web 数据库开发者提供完整的数据库解决方案。ADO 可以与 ASP 紧密结合，运行在服务器端，从而制作出功能完善的数据库网页。ADO 除了支持各种数据库平台外，它还支持 VB、VC 和 Script 等多种程序设计语言。

在利用 ADO 技术进行数据库网页制作的第一步就是要建立服务器端的数据库，即后台数据库，一般可选择 SQL Server 或者 Access 作为后台数据库。本次的毕业设计是以 SQL Server 作为后台数据库。

第 3 章 需求分析

3.1 系统主要工作流程图

3.1.1 后台管理模块流程图

该流程图描述了后台管理员对的管理，管理员可以对商品进行添加，修改，删除。具体的添加功能是通过本地浏览把商品上传到；修改商品信息或者删除商品等功能,如图 3-1 所示。

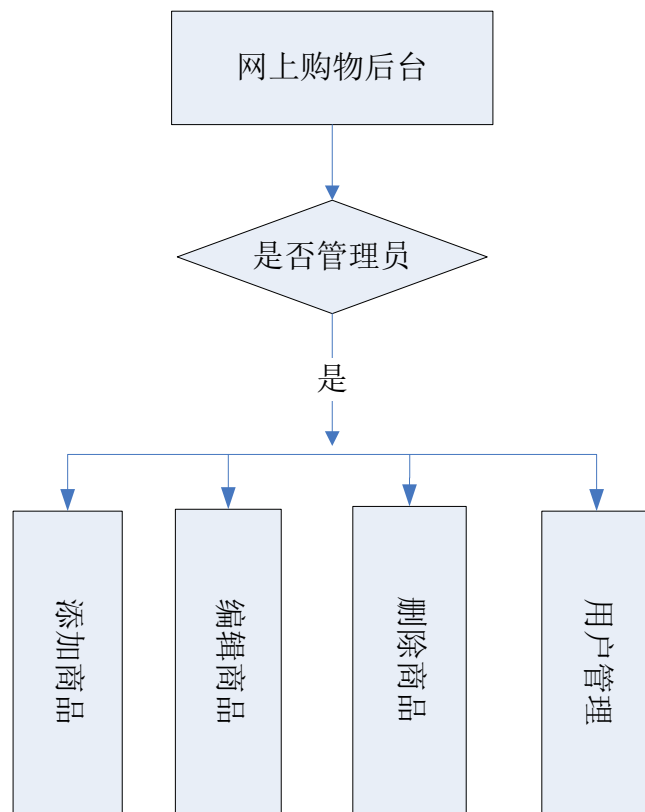


图 3-1 后台管理程序主要流程图

3.1.2 前台管理模块流程图

本系统设计的初衷是会员可以享受的功能，未注册的游客只能浏览，不能进行其他操作。因此当用户在进入的一开始便会对用户的身份和权限进行验证，反馈不同的页面给不同的用户，从而实现对权限的划分功能。可在上在线购买商品、搜索商品等功能。该流程图描述了商品购物

的整个流程，如图 3-2 所示。

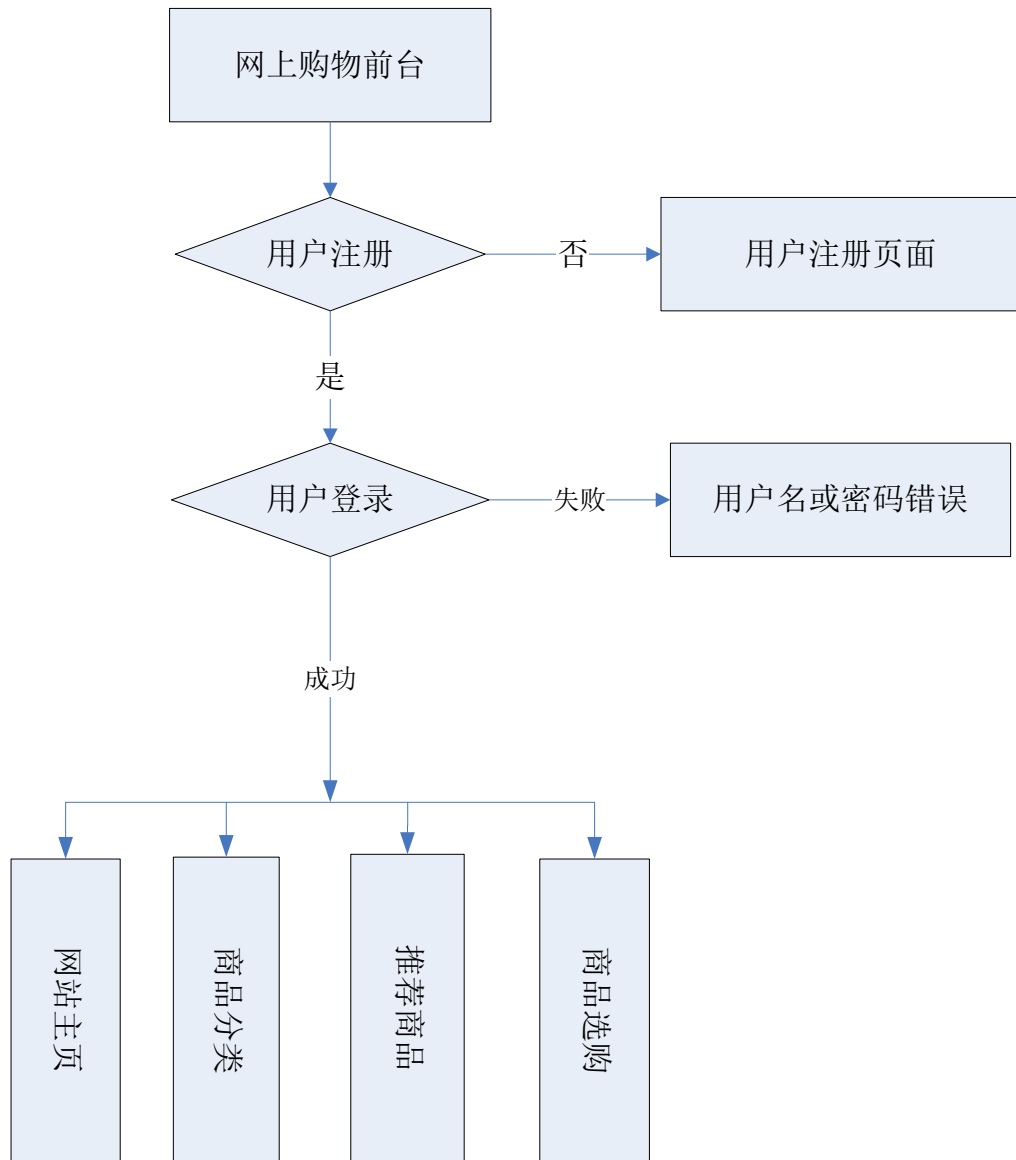


图 3-2 前台管理程序主要流程图

3.2 系统实现采用环境

1. 操作系统：Microsoft Windows XP Professional。
2. 开发语言：JAVA。
3. 数据库系统：SQL Server。
4. 开发平台：MyEclipse。

3.3 系统功能需求

操作的简易实用性本系统是网上商品购物系统,针对的用户社会上所有的人们。因此操作的简易实用性就体现的尤其重要。在此系统的开发中就很好的体现了这一点,系统的界面美观,典雅,充满了人性化;用户操作起来也容易上手。对于一个网上商品购物系统而言,网上的商品的种类样式是很重要的,所以对系统的安全性有比较高的要求:对于数据库,要设置不同用户的权限,数据的修改必须由合法用户操作。

第 4 章 功能的设计与实现

4.1 系统主要设计

4.1.1 系统主要功能

1. 用户注册登录功能：只有登录注册后才能使用本功能；
2. 购物车：用来存放客户购物过程中选出的货物；
3. 浏览最新商品品信息：了解到最新商品新闻；
4. 搜索商品：搜索客户想要的商品；
5. 商品管理：后台管理员可以增、删、编辑商品信息；

4.1.2 主要功能模块

1. 后台管理模块

该模块只对管理员开放，管理员可以对商品进行添加，修改，删除。具体的添加功能是通过浏览把商品上传到；修改商品信息或者删除商品等功能。如图 4-1 所示。

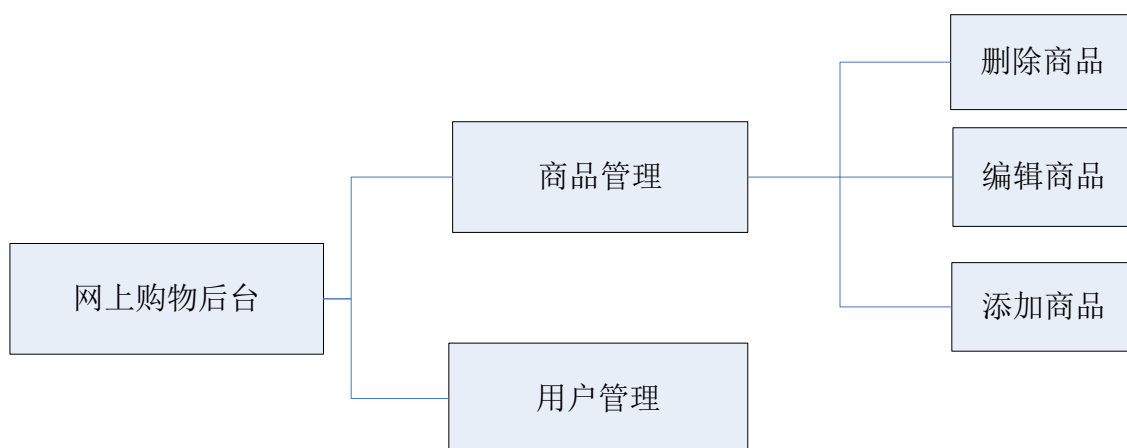


图 4-1 后台管理模块

2. 前台管理模块

该模块用户注册后即成为会员，就可在线上在线购买商品、搜索商品等功能。详细模块的介绍如图 4-2 所示。

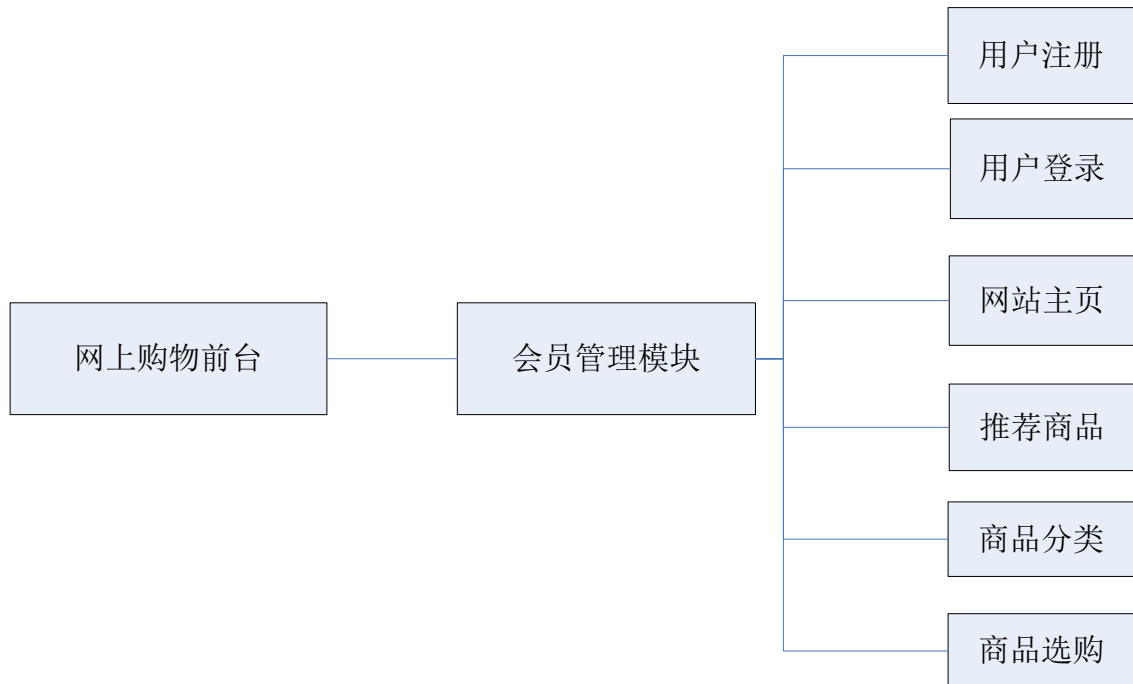


图 4-2 前台管理模块图

4.2 数据库的分析与设计

4.2.1 数据库的需求分析

本课题开发的数据库主要包括四个方面。

1. 会员注册数据表：存储会员注册时各种信息和账户的激活，充值，收货地址等信息的数据表。
2. 商品数据表：商品基本参数信息，包括商品的出处，商品的生产日期，商品的性能等的数据表。
3. 供货商数据表：供货商的基本参数信息，包括供货商的名称，地址，联系方式等的数据表。
4. 订单信息数据表：用户购买商品的基本参数信息，包括订单号，收货人的基本信息，配送地址等的数据表。

4.2.2 数据库的结构设计

会员信息表(TBa_MemberInfo):

该表有递增的流水号, 用户名, 密码,, 余额, 注册日期, 激活日期, 备注等。

表 4-1 会员信息表

字段名称	含义	数据类型	主键
ID	递增的流水号	numeric	是
UserName	用户名	varchar(20)	否
Pwd	密码	varchar(20)	否
Email	Email	varchar(50)	否
lName		varchar(20)	否
Balance	余额	money	否
Status	状态	bit	否
RegDate	注册日期	datetime	否
ActiveDate	激活日期	datetime	否
Remark	备注	varchar(100)	否

会员充值表(TBa_SupplyRecordInfo):

该表有用户名, 付款账号, 付款开户行, 收款账号, 收款开户行等。

表 4-2 会员充值表

字段名称	含义	数据类型	主键
ID	递增的流水号	numeric	是
UserName	用户名	varchar(20)	否
PayAccountNo	付款账号	varchar(19)	否
PayBank	付款开户行	varchar(50)	否
RecAccountNo	收款账号	varchar(19)	否
RecBank	收款开户行	varchar(50)	否
Remark	备注	varchar(50)	否
TotalMoney	金额	money	否
SupplyTime	充值时间	datetime	否

以上内容仅为本文档的试下载部分,为可阅读页数的一半内容。
如要下载或阅读全文,请访问:

<https://d.book118.com/518013003047006120>