

同济大学计算机科学与技术系

软件开发方法课程期末项目报告



学 号 1853971

姓 名 王天

专 业 计算机科学与技术

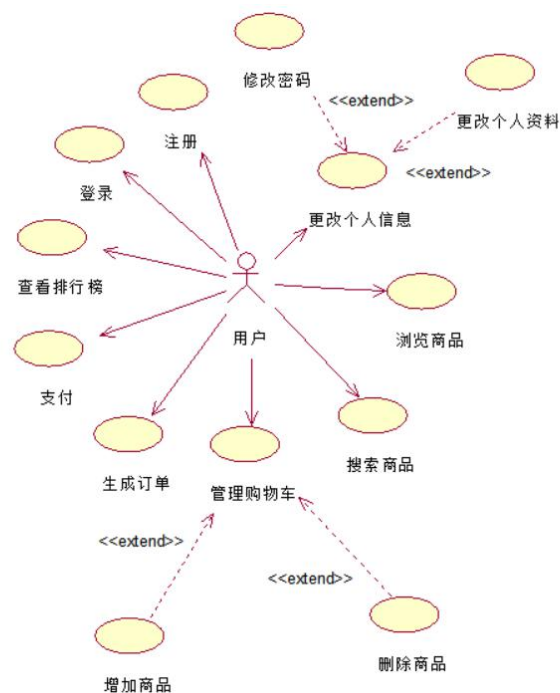
任课教师 王小平

日 期 2020-06-11

项目名称 基于 Web 的电子商城系统分析与设计

一.用例图与用例说明

1.用户参与的用例图



角色说明：

角色：用户

角色职责：使用电子商城系统的购物服务

角色职责识别：注册和登录、商品列表显示、商品详细信息查看、用户购物车管理、订单处理、浏览排行榜、支付

用例说明：

(1)用例名称：注册

用例描述：点击注册按钮，提供必要的身份信息及联系方式后成为会员

参与者：用户

前置条件：进入电子商城网站并且尚未登录

主要步骤：

- ① 用户点击注册按钮
- ② 用户输入身份信息和联系方式
- ③ 用户完成邮箱/手机号验证
- ④ 注册成功

后置条件：用户获得用于登录电子商城系统的会员账号

相关用例：登录

(2)用例名称：登录

用例描述：用户登录会员账号以开始使用电子商城系统功能

参与者：用户

前置条件：进入电子商城网站并且尚未登录但拥有会员账号

主要步骤：

- ① 用户点击登录按钮
- ② 用户输入账号信息和密码
- ③ 验证账号信息和密码是否匹配
- ④ 登录成功

后置条件：用户可以使用电子商城的商品列表显示、商品详细信息查看、用户购物车管理、订单处理、浏览排行榜、支付等功能。

相关用例：注册、商品列表显示、商品详细信息查看、用户购物车管理、订单处理、浏览排行榜、支付

(3)用例名称：更改个人信息

用例描述：用户修改会员账号信息

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

- ① 用户点击更改个人信息按钮
- ② 用户编辑个人信息
- ③ 用户保存个人信息
- ④ 更改个人信息完成

后置条件：无

相关用例：修改密码用例、更改个人资料用例

(4)用例名称：修改密码

用例描述：用户修改会员账号密码

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

- ① 用户点击更改个人信息按钮
- ② 用户编辑新密码
- ③ 用户确认新密码
- ④ 修改密码完成

后置条件：无

相关用例：更改个人信息、更改个人资料用例

(5)用例名称：更改个人资料

用例描述：用户更改个人资料

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

- ① 用户点击更改个人信息按钮
- ② 用户编辑个人资料
- ③ 用户保存个人资料
- ④ 更改个人资料完成

后置条件：无

相关用例：更改个人信息、修改密码

(6)用例名称：浏览商品

用例描述：查看电子商城系统商品的信息

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

① 用户进入商品区域

② 用户点击商品图片或链接

后置条件：可以将商品加入购物车

相关用例：搜索商品、管理购物车

(7)用例名称：搜索商品

用例描述：根据商品名称进行模糊搜索，查找相关商品信息。

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

① 用户点击搜索框

② 用户编辑关键词

③ 用户点击搜索按钮

④ 显示搜索商品的结果

后置条件：可以浏览商品或加入购物车

相关用例：浏览商品、管理购物车

(8)用例名称：查看排行榜

用例描述：查看电子商城系统首页的“浏览排行榜”和“销量排行榜”来获取热门商品信息。

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

① 用户点击“浏览排行榜”或“销量排行榜”按钮

② 显示商品排行榜信息

后置条件：可以浏览商品或加入购物车

相关用例：浏览商品、管理购物车

(9)用例名称：管理购物车

用例描述：管理购物车中的商品数量和种类

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

① 用户点击“购物车”按钮

② 用户选择删除或添加商品

③ 管理购物车结束

后置条件：可以生成购物订单，购买商品

相关用例：增加商品用例、删除商品用例、生成订单用例、支付用例

(10)用例名称：增加商品

用例描述：增加购物车中的商品

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

- ① 用户点击“购物车”按钮
- ② 用户选择添加商品
- ③ 增加商品结束

后置条件：可以生成购物订单，购买商品

相关用例：管理购物车用例、删除商品用例、生成订单用例、支付用例

(11)用例名称：删除商品

用例描述：删除购物车中的商品

参与者：用户

前置条件：进入电子商城网站并且已经登录

主要步骤：

- ① 用户点击“购物车”按钮
- ② 用户选择删除商品
- ③ 删除商品结束

后置条件：可以生成购物订单，购买商品

相关用例：管理购物车用例、增加商品用例、生成订单用例、支付用例

(12)用例名称：生成订单

用例描述：根据购物车中的商品生成商品订单信息

参与者：用户

前置条件：进入电子商城网站并且已经登录、购物车中有商品

主要步骤：

- ① 用户点击“购物车”按钮
- ② 用户选择生成订单
- ③ 用户确认订单信息
- ④ 生成订单完成

后置条件：可以支付并购买商品

相关用例：管理购物车用例、支付用例

(13)用例名称：支付

用例描述：根据订单信息支付相应金额完成购物交易

参与者：用户

前置条件：已经生成订单

主要步骤：

- ① 用户点击“立即支付”按钮
- ② 用户选择支付手段

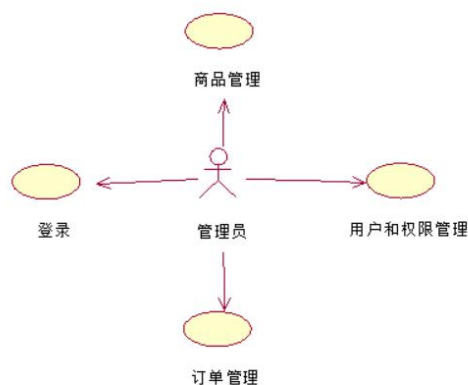
③ 用户确认支付

④ 支付完成

后置条件：无

相关用例：生成订单用例

2.管理员参与的用例图



角色说明：

角色名称：管理员

角色职责：管理电子商城网站事务

角色职责识别：商品管理、用户和权限管理、订单管理

用例说明：

(1)用例名称：登录

用例描述：管理员登录电子商城后台系统

参与者：管理员

前置条件：拥有管理员账号

主要步骤：

- ① 管理员点击“登录”按钮
- ② 管理员提供账号信息和密码
- ③ 验证账号信息和密码是否匹配
- ④ 登录完成

后置条件：管理员可以进行对电子商城系统后台的管理功能

相关用例：商品管理用例、订单管理用例、用户和权限管理用例

(2)用例名称：商品管理

用例描述：管理员对电子商城中的商品进行管理

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“商品管理”按钮
- ② 管理员选择具体商品
- ③ 管理员修改商品信息

④ 商品管理完成

后置条件：无

相关用例：显示商品列表用例、修改商品用例、查询商品用例、删除商品用例、修改商品类型用例、删除商品类型用例、显示商品类型列表用例

(3)用例名称：订单管理

用例描述：管理员对电子商城中的订单进行管理

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

① 管理员点击“订单管理”按钮

② 管理员选择具体订单

③ 管理员修改订单信息

④ 订单管理完成

后置条件：无

相关用例：创建订单用例、删除订单用例、查询订单用例、修改订单用例、查看订单明细用例

(4)用例名称：用户和权限管理用例

用例描述：管理员对电子商城中的用户和管理员的权限进行管理

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

① 管理员点击“用户管理”或“权限”按钮

② 管理员选择具体用户/管理员

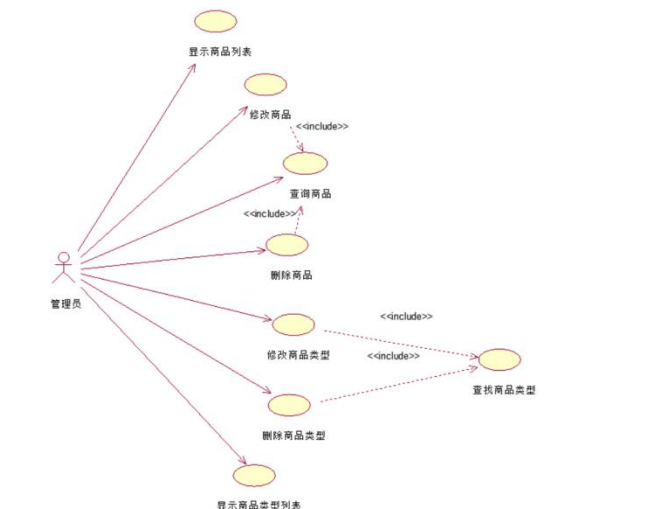
③ 管理员修改用户/管理员权限信息

④ 用户和管理员权限管理完成

后置条件：无

相关用例：查看用户列表用例、禁用用户用例、启用用户用例、新增管理员用例、设置管理员权限用例

3.管理员的商品管理用例的拆分



(1)用例名称：显示商品列表用例

用例描述：管理员对电子商城中的商品信息进行总览

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“商品列表”按钮
- ② 管理员选择查看的商品条件
- ③ 显示电子商城中的所有满足条件的商品的列表

后置条件：对具体商品进行管理

相关用例：修改商品用例、查询商品用例、删除商品用例

(2)用例名称：修改商品用例

用例描述：管理员对电子商城中的商品信息进行总览

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理商品”按钮
- ② 管理员选择具体商品
- ③ 管理员修改对应商品的信息

后置条件：无

相关用例：查询商品用例、删除商品用例

(3)用例名称：查询商品用例

用例描述：管理员对电子商城数据库中的商品信息进行查询

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理商品”按钮
- ② 管理员输入相关条件
- ③ 管理员查询对应商品的信息(包括库存数量、商品价格、商品状态等)

后置条件：无

相关用例：修改商品用例、删除商品用例

(4)用例名称：删除商品用例

用例描述：管理员对电子商城数据库中的商品信息进行删除

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理商品”按钮
- ② 管理员查询具体商品
- ③ 管理员删除对应商品的信息

后置条件：可以对商品进行修改和删除

相关用例：修改商品用例、查询商品用例

(5)用例名称：修改商品类型用例

用例描述：管理员对电子商城数据库中的商品类型信息进行修改

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理商品”按钮
- ② 管理员查找具体商品类型
- ③ 管理员修改对应商品类型的信息

后置条件：无

相关用例：显示商品类型用例、删除商品类型用例

(6)用例名称：删除商品类型用例

用例描述：管理员对电子商城数据库中的商品类型信息进行删除

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理商品”按钮
- ② 管理员查找具体商品类型
- ③ 管理员删除对应商品类型的信息

后置条件：无

相关用例：修改商品类型用例、显示商品类型用例

(7)用例名称：显示商品类型列表用例

用例描述：管理员对电子商城数据库中的商品类型信息进行总览

参与者：管理员

前置条件：已经登录电子商城系统后台

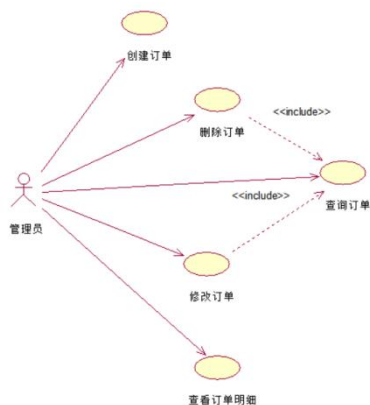
主要步骤：

- ① 管理员点击“管理商品”按钮
- ② 管理员选择查看的商品类型的条件
- ③ 显示电子商城系统中满足条件的所有商品类型的列表

后置条件：可以对商品进行修改和删除

相关用例：修改商品类型用例、删除商品类型用例

4.管理员的订单管理用例的拆分



(1)用例名称：创建订单用例

用例描述：管理员向电子商城数据库中添加订单信息

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理订单”按钮
- ② 管理员点击“添加订单”按钮
- ③ 管理员编辑订单内容
- ④ 管理员保存订单内容
- ⑤ 订单创建完成

后置条件：无

相关用例：查看订单明细用例、删除订单用例、查询订单用例、修改订单用例

(2)用例名称：查询订单用例

用例描述：管理员在电子商城数据库中查询订单信息

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理订单”按钮
- ② 管理员提供查询的订单的条件
- ③ 显示满足条件的订单

后置条件：可以对订单进行删除、修改和查看详细内容

相关用例：查看订单明细用例、删除订单用例、修改订单用例

(3)用例名称：删除订单用例

用例描述：管理员在电子商城数据库中删除订单信息

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理订单”按钮
- ② 管理员提供查询的订单的条件
- ③ 管理员删除具体的的订单

后置条件：无

相关用例：查看订单明细用例、查询订单用例、修改订单用例

(4)用例名称：修改订单用例

用例描述：管理员在电子商城数据库中修改订单信息

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理订单”按钮
- ② 管理员提供查询的订单的条件
- ③ 管理员修改具体的的订单

后置条件：无

相关用例：查看订单明细用例、查询订单用例、删除订单用例

(5)用例名称：查看订单明细用例

用例描述：管理员在电子商城数据库中查看订单的具体内容

参与者：管理员

前置条件：已经登录电子商城系统后台

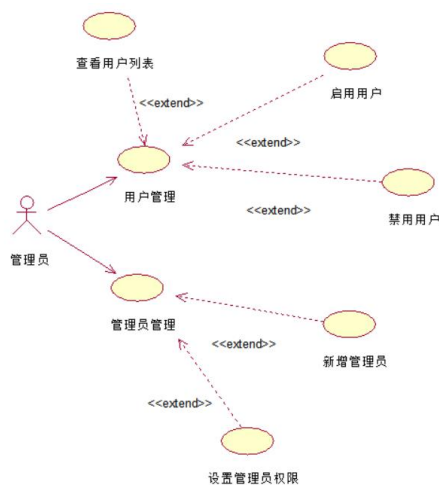
主要步骤：

- ① 管理员点击“管理订单”按钮
- ② 管理员提供查询的订单的条件
- ③ 管理员查看订单的具体内容

后置条件：无

相关用例：删除订单用例、查询订单用例、修改订单用例

5.管理员的用户和权限管理用例的拆分



(1)用例名称：查看用户列表用例

用例描述：管理员在电子商城数据库中查看注册的会员的总览

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理用户”按钮
- ② 管理员提供要查看的会员的条件
- ③ 显示满足条件的电子商城系统数据库中的所有会员的列表

后置条件：可以对具体的用户进行管理

相关用例：禁用用户用例、启用用户用例

(2)用例名称：禁用用户用例

用例描述：管理员在电子商城数据库中禁用会员的使用功能

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理用户”按钮

- ② 管理员提供要禁用的会员的条件
 - ③ 管理员禁用对应的用户
- 后置条件：被禁用的用户无法正常使用商城功能
- 相关用例：查看用户列表用例、启用用户用例

(3)用例名称：启用用户用例

用例描述：管理员在电子商城数据库中启用会员的使用功能

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“管理用户”按钮
- ② 管理员提供要启用的会员的条件
- ③ 管理员启用对应的用户

后置条件：被启用的用户可以正常使用商城功能

相关用例：查看用户列表用例、禁用用户用例

(4)用例名称：新增管理员用例

用例描述：管理员在电子商城数据库中新增管理员信息

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“新增管理员”按钮
- ② 管理员编辑新的管理员的信息
- ③ 管理员保存新增管理员信息
- ④ 新增管理员完成

后置条件：无

相关用例：设置管理员权限用例

(5)用例名称：设置管理员权限用例

用例描述：管理员在电子商城数据库中设置管理员权限

参与者：管理员

前置条件：已经登录电子商城系统后台

主要步骤：

- ① 管理员点击“权限管理”按钮
- ② 管理员选择要设置权限的管理员
- ③ 管理员重新设置具体管理员的权限
- ④ 管理员保存新设置的权限
- ⑤ 设置管理员权限完成

后置条件：无

相关用例：新增管理员用例

二.数据模型和数据库表设计、E-R 图

在数据库中创建 8 张表用以描述系统所需要用到的数据。

1. 用户信息表 user_info，用于记录前台用户基本信息

字段名	数据类型	大小	是否为空	主外键	说明
user_id	int	4	N	PK	会员 id 标识，主键，自增
user_name	varchar	16	N		会员账号
user_password	varchar	16	N		会员密码
realName	varchar	16	Y		真实姓名
sex	varchar	4	Y		会员性别
address	varchar	255	Y		联系地址
question	varchar	50	Y		验证问题
answer	varchar	50	Y		验证问题的答案
email	varchar	50	Y		电子邮件
favorite	varchar	50	Y		个人爱好
score	int	4	Y		等级
regDate	date		Y		注册时间

2. 管理员信息表 admin_info，用于记录管理员基本信息

字段名	数据类型	大小	是否为空	主外键	说明
admin_id	int	4	N	PK	管理员 id 标识，主键，自增
admin_name	varchar	16	Y		管理员姓名
pwd	varchar	16	Y		管理员密码
role	varchar	8	Y		权限

3. 商品类型表 type，用于记录各种商品类型

字段名	数据类型	大小	是否为空	主外键	说明
type_id	int	4	N	PK	类型 id 标识，主键，自增
type_name	varchar	20	Y		类型名称

4. 商品信息表 product_info，用于记录商品信息

字段名	数据类型	大小	是否为空	主外键	说明
product_id	int	4	N	PK	商品 id 标识，主键，自增
code	varchar	16	N		商品编号
product_name	varchar	255	Y		商品名称
tid	int	4	Y	FK	商品类别 id
brand	varchar	20	Y		品牌

pic	varchar	255	Y		商品图片
product_num	int	4	Y		商品数量
price	decimal	10,0	Y		商品价格
intro	longtext		Y		商品介绍
product_status	int	4	Y		商品状态
bigpic	varchar	255	Y		商品大图片

5. 订单信息表 order_info，用于记录订单主要信息

字段名	数据类型	大小	是否为空	主外键	说明
order_id	int	4	N	PK	订单 id 标识，主键，自增
uid	int	4	N	FK	用户 id
order_status	varchar	16	Y		订单状态
order_time	datetime		Y		订单下单时间
order_price	decimal	8,2	Y		订单价格

6. 订单明细表 order_detail

字段名	数据类型	大小	是否为空	主外键	说明
order_detail_id	int	4	N	PK	订单明细 id 标识，主键，自增
oid	int	4	N	FK	订单 id
pid	int	4	N	FK	商品 id
num	int	4	Y		购买数量

7. 系统功能表 functions

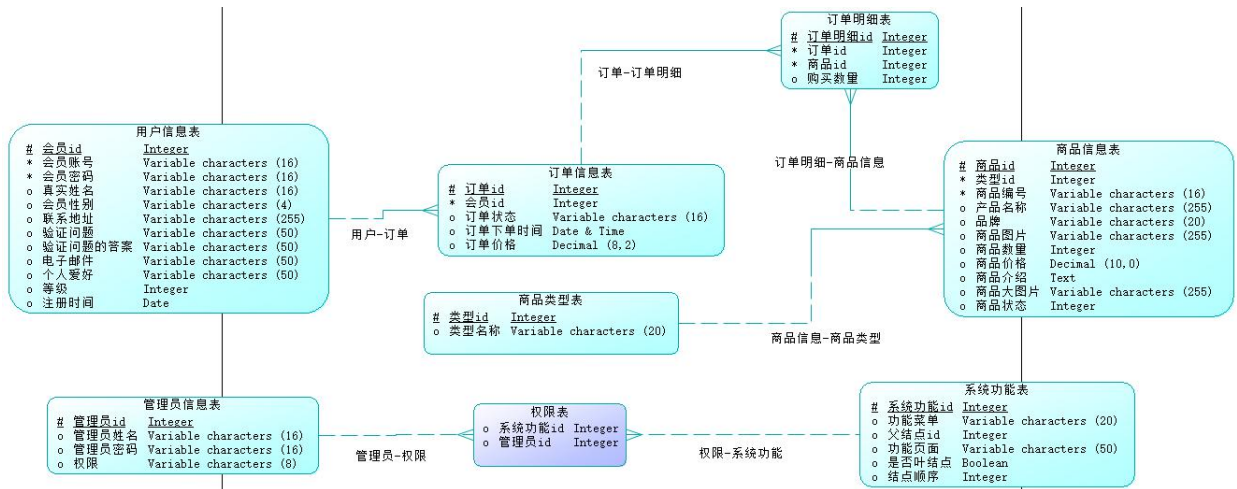
字段名	数据类型	大小	是否为空	主外键	说明
function_id	int	4	N	PK	系统功能 id 标识，主键，自增
function_name	varchar	20	Y		功能菜单
parentid	int	4	Y		父结点 id
url	varchar	50	Y		功能页面
isleaf	bit	1	Y		是否叶结点
nodeorder	int	4	Y		结点顺序

8. 权限表 powers

字段名	数据类型	大小	是否为空	主外键	说明
aid	int	4	N	PK	管理员 id，主键
fid	int	4	N	FK	系统功能 id

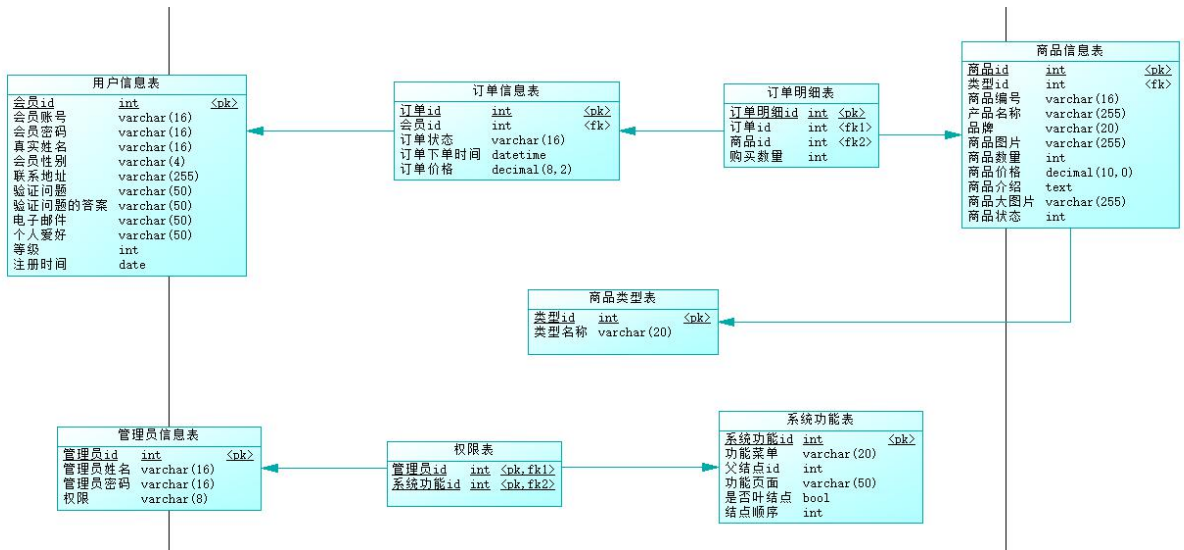
根据 8 张表之间的逻辑关系，画出 E-R 图如下所示：

① 逻辑模型：



图中每一条连线代表了一个关系，其中带多各分支的一端代表对应端的对象可以有多个；实线端代表该端对象不能离开另一端的对象，即另一端的对象至少有 1 个；虚线端代表该端对象可以离开另一端对象，即另一端的对象可以是 0 个。

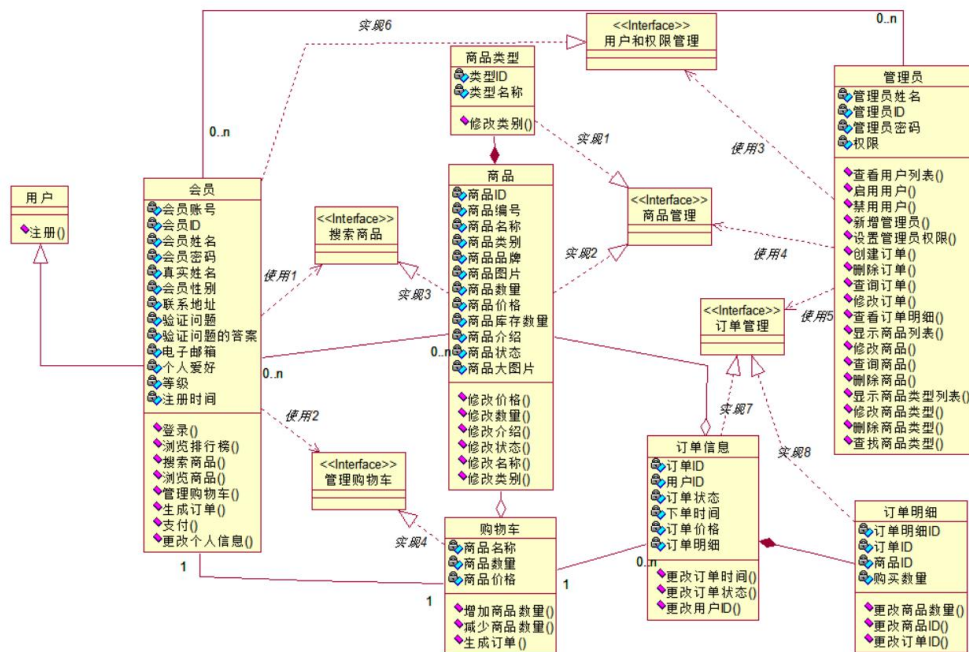
② 物理模型：



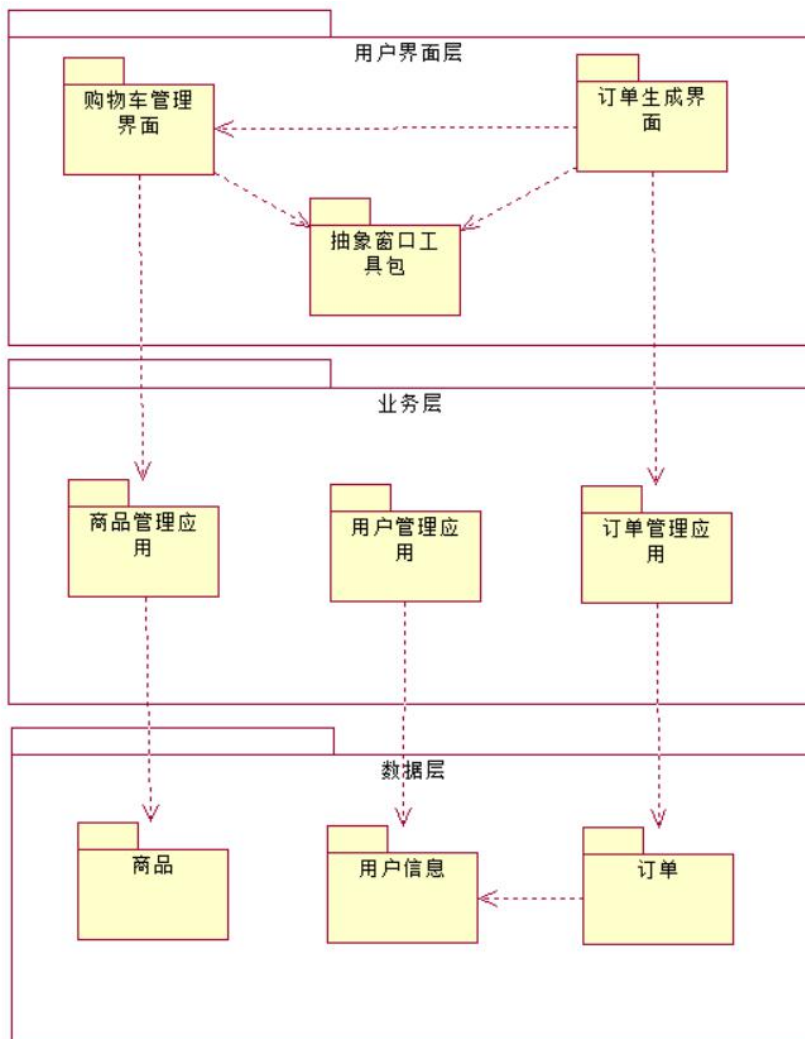
三、类、类图、包图的设计

使用系统实体识别法识别出以下类：用户、会员、管理员、商品、商品类型、购物车、订单信息、订单明细，并且确定各类所具有的方法和属性。

使用类图描述类之间的关系。通过接口实现会员/管理员的各项权限。



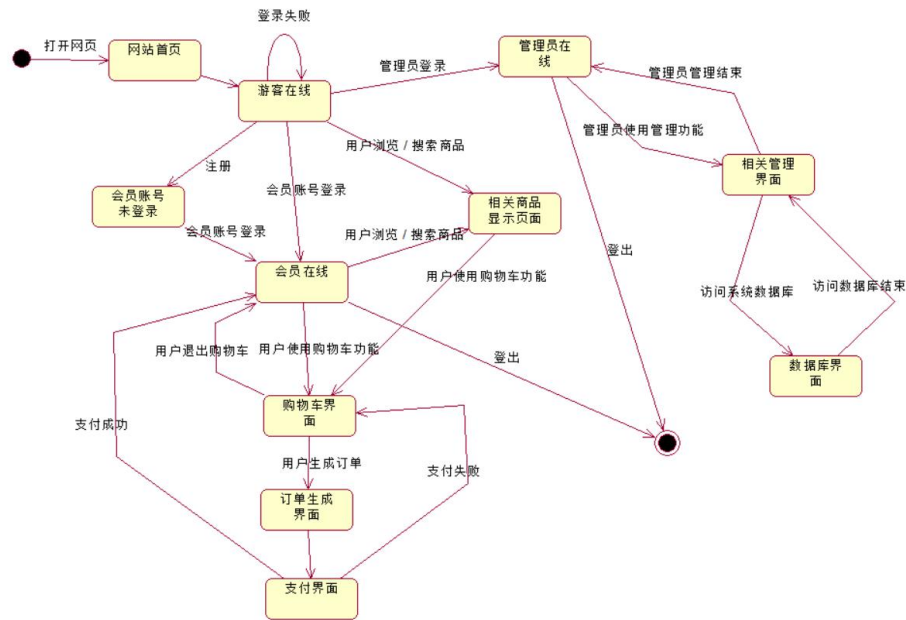
整个电子商城系统应该包含三个包:界面包(用户接口包)、业务包(系统应用)和数据包(问题域包), 并且各自包含子包, 画出三层结构的包图表示各个包之间的关系如下:



四.系统动态模型设计

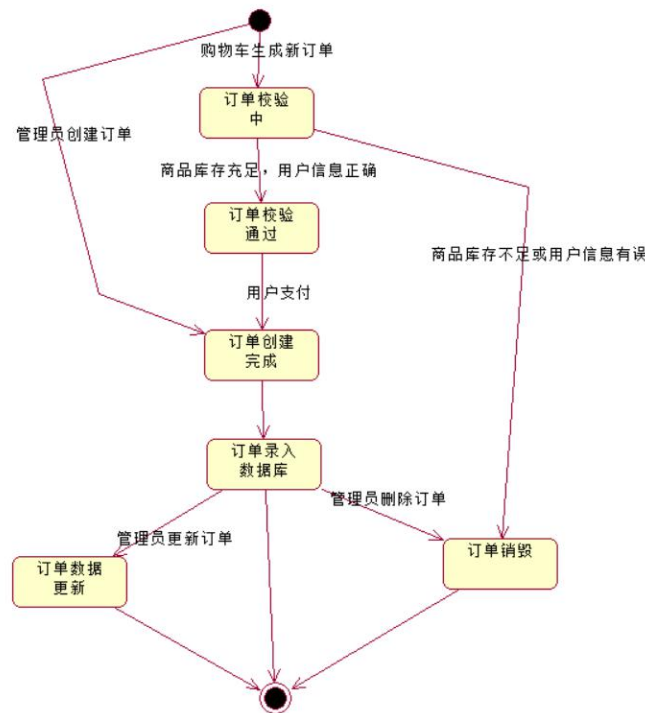
1.状态图

(1)电子商城系统网页的状态图



整个电子商城随着用户的注册和登录、浏览不同的商品、进入购物车、生成订单和支付、管理员登录以及进行不同权限的管理等操作而在不同页面之间跳转。

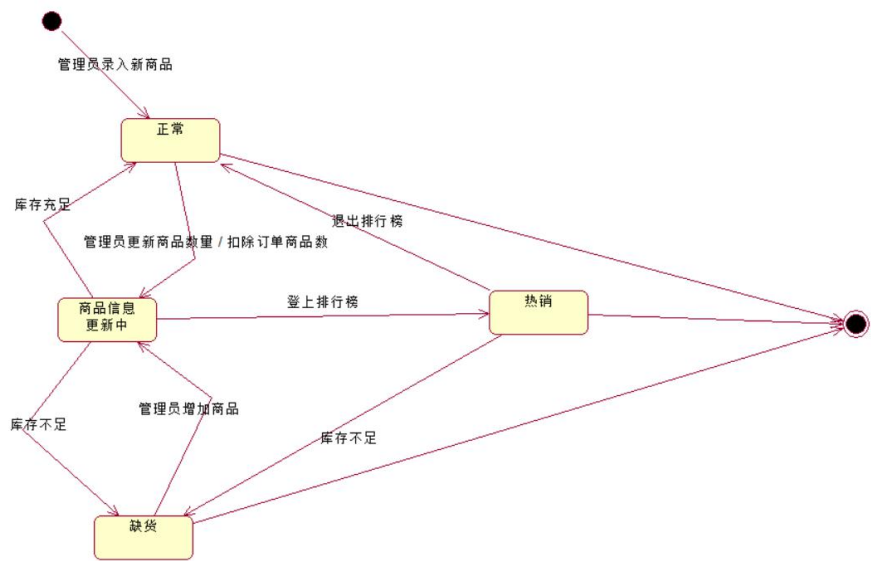
(2)订单的状态图



订单由购物车生成，在正式提交前需要检查商城的商品库存，并由会员确认订单信息，

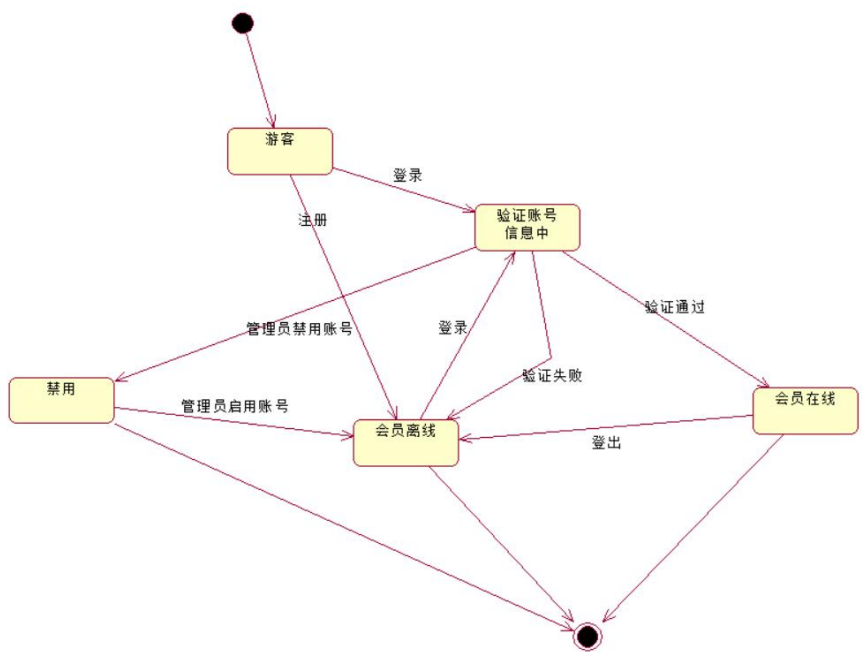
在会员支付成功后才创建完成。管理员从后台能够对订单信息做出管理。

(3)商品的状态图



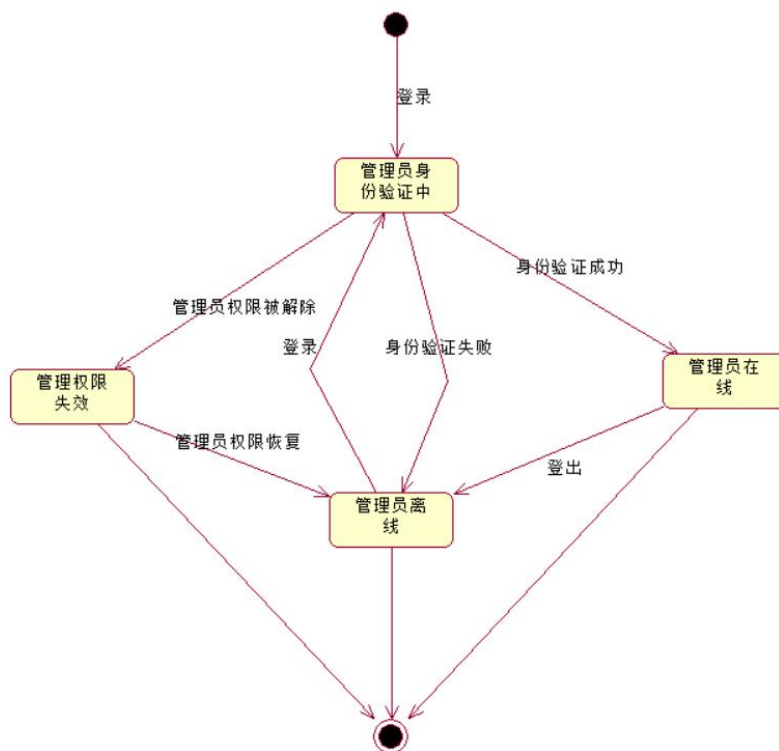
商品的状态包括正常、热销和缺货。当商城的商品库存不足时，对应商品就会进入缺货状态。当商品浏览量或销量过高时则可以在排行榜上进行显示。

(4)用户的状态图



用户的状态随着注册/登录等功能的使用而改变，并且管理员可以通过后台禁用/启用用户以限制/恢复用户对电子商城功能的正常使用。

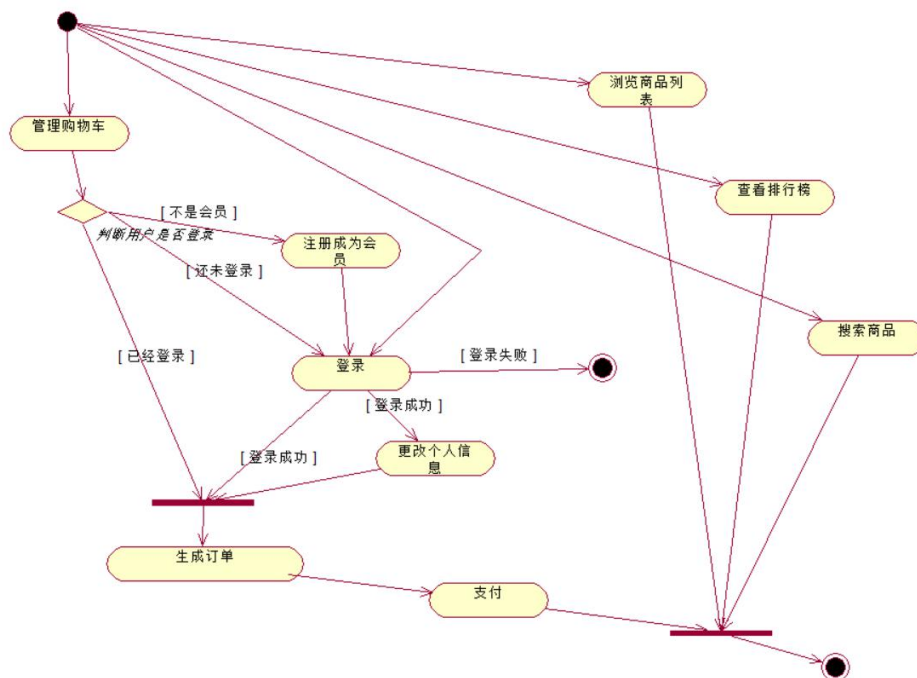
(5)管理员的状态图



管理员的状态随着管理员登录等功能的使用而变化,并且可由更高等级的管理员对管理权限做出更改,可能获得/丧失管理员权限。

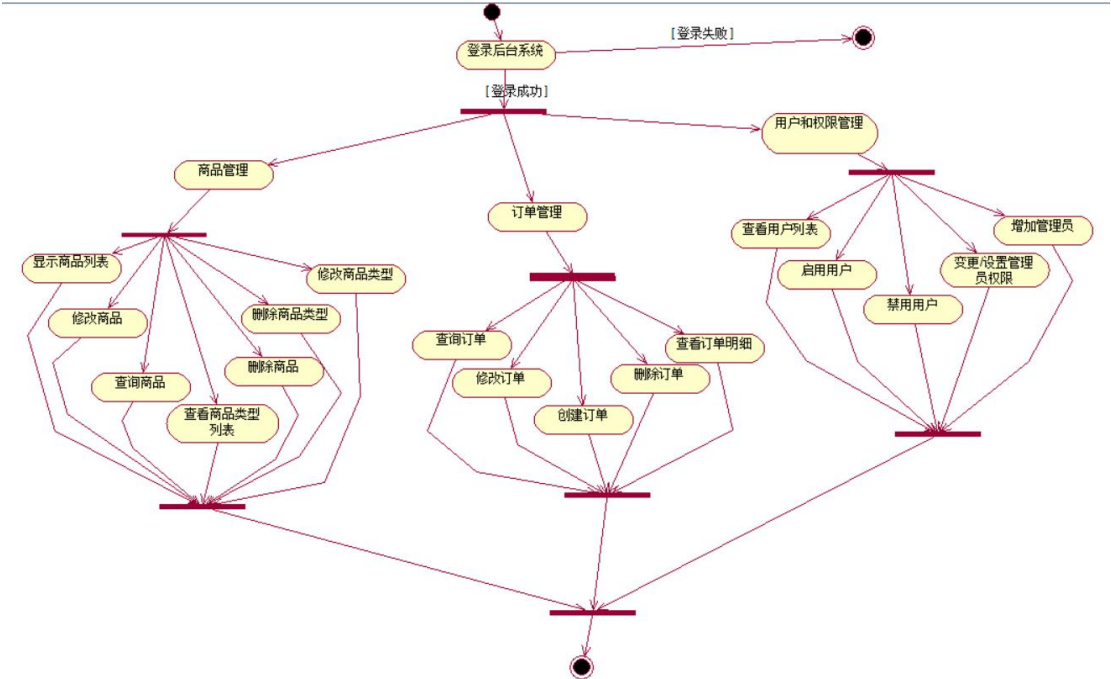
2.活动图

(1)用户使用电子商城系统的活动图



用户需要以会员身份登录才能使用购物车功能,使用时若是游客状态则会被电子商城系统要求登录。只有从购物车才能生成订单并支付,完成购物交易。对商品的浏览、搜索和排行榜的查看等功能在游客状态也可以使用。

(2)管理员使用电子商城管理系统的活动图

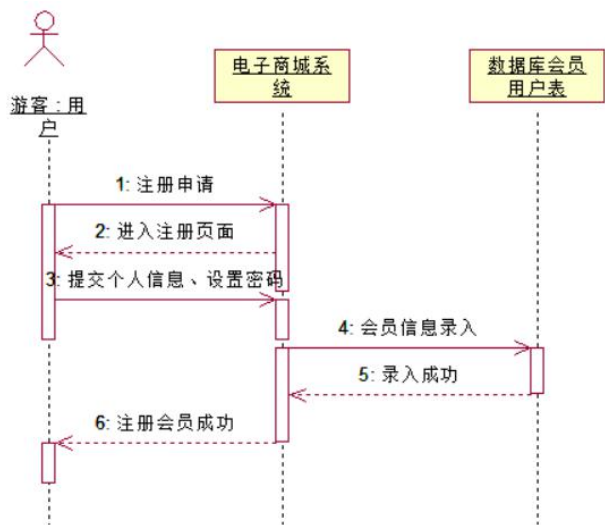


管理员需要登录后台系统才能开始对电子商城系统的管理。获得了相应的权限之后,管理员可以分别进行商品管理(包括显示商品列表、修改商品、查询商品、删除商品、查看商品类型列表、删除商品类型、修改商品类型)、订单管理(包括查询订单、修改订单、创建订单、删除订单、查看订单明细)、用户和权限管理(包括查看用户列表、启用用户、禁用用户、增加管理员、设置管理员权限)。

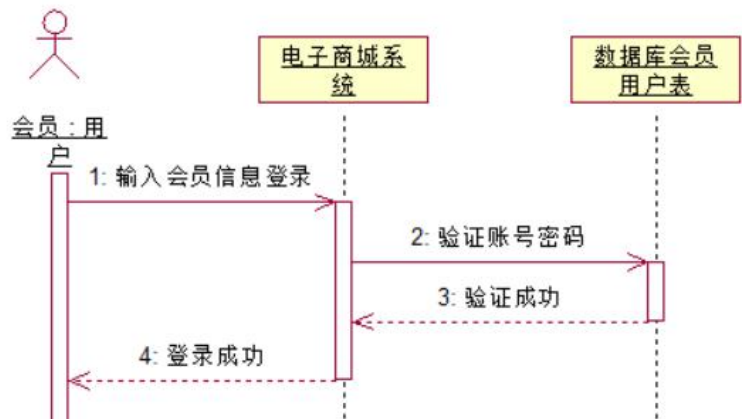
3.顺序图

(1)前台功能的动态模型设计:

①游客注册成为会员以及会员登录:

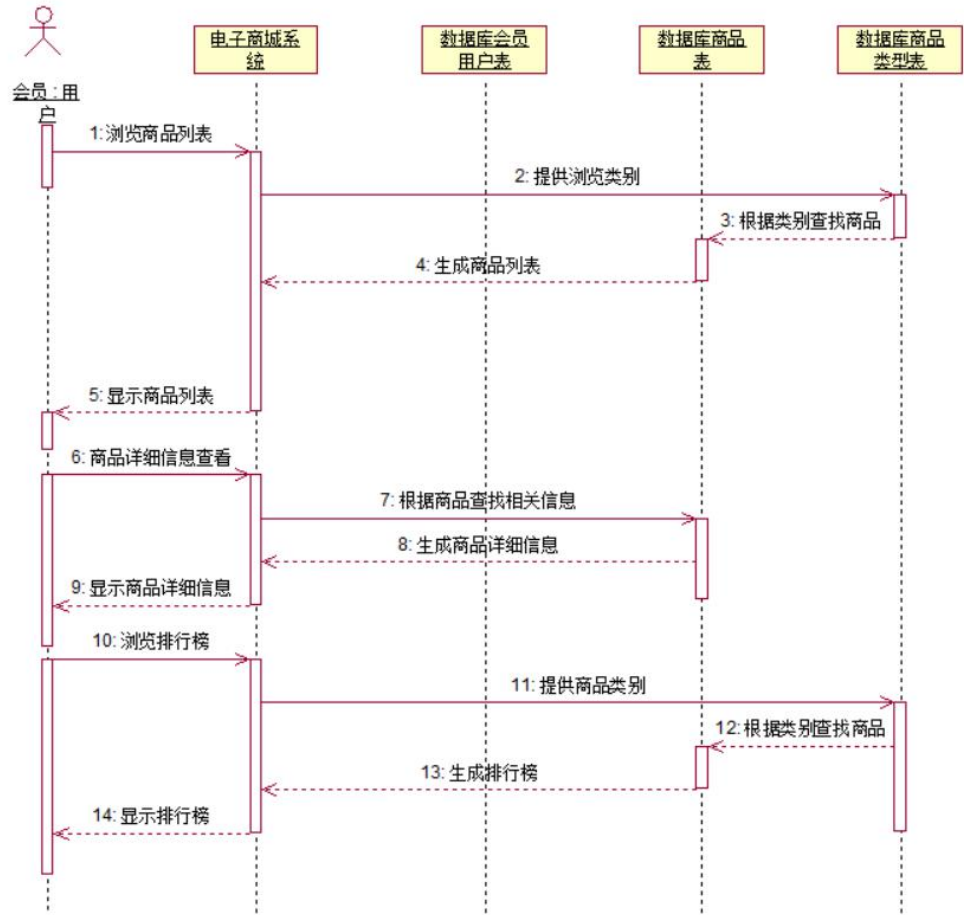


游客提出注册申请,进入注册页面,提交个人信息并设置密码之后由系统录入会员信息,完成游客的会员注册。



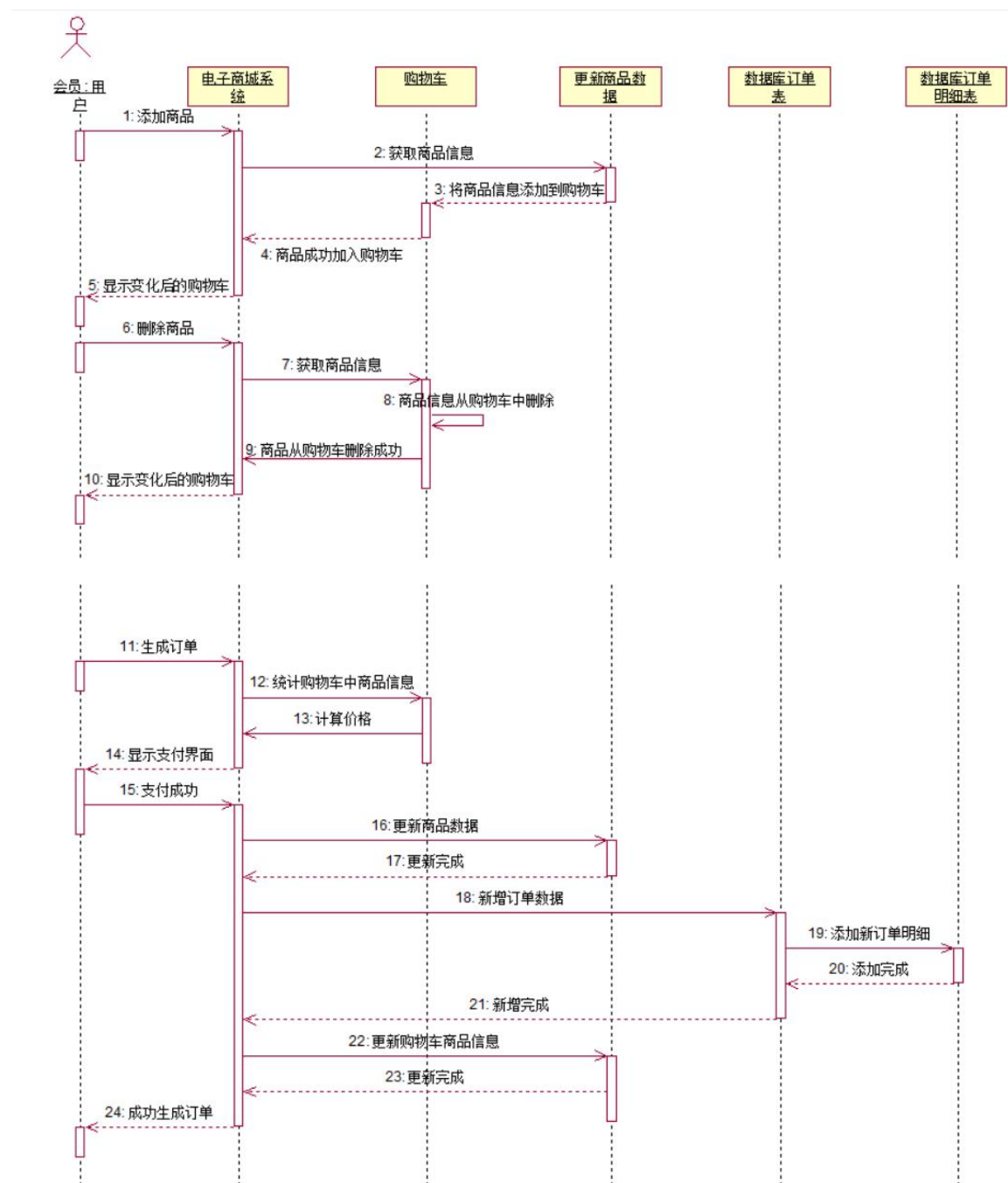
会员输入账号、密码信息，由系统验证通过后登录成功。

③ 会员浏览商品列表、查看商品信息与浏览排行榜



用户浏览商品时可以提供浏览类别或者默认，以查看商品列表。在列表中点击商品，系统将会生成商品的详细信息并显示。用户可以提供类别以查看对应商品类型的排行榜。

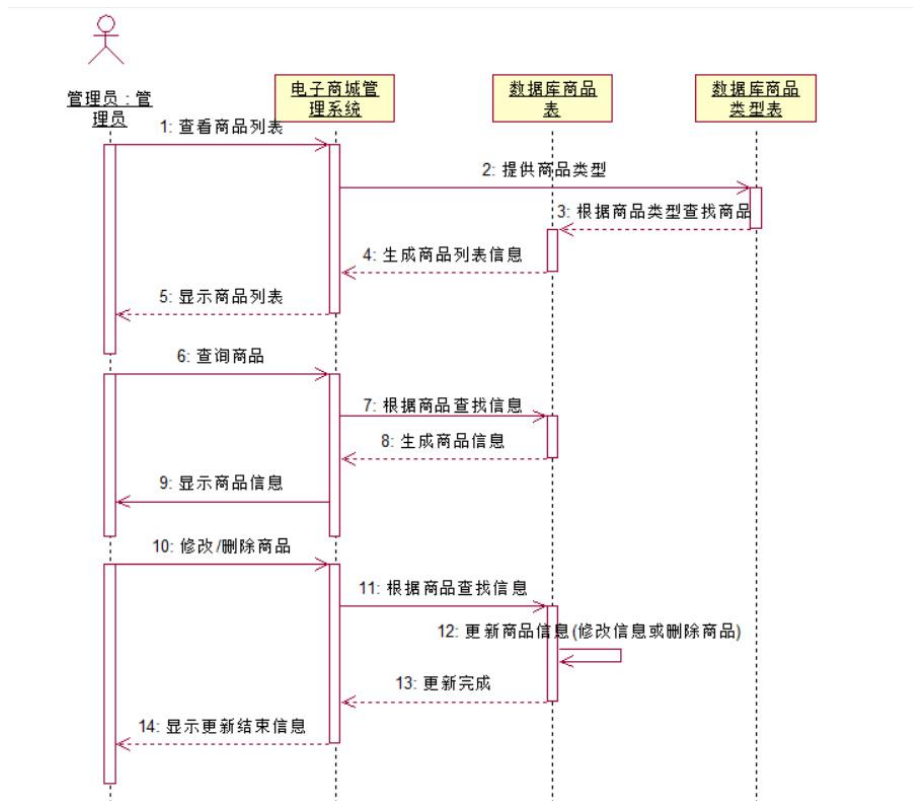
④ 用户管理购物车和生成订单



用户向购物车中添加/删除商品时，系统更新购物车中数据，并显示购物车的变化。用户生成订单时，由系统计算订单价格并要求用户进行支付，支付完成后更新系统的商品库存等信息，并添加新的订单信息，最后更新购物车中数据，减去订单中购买的商品。

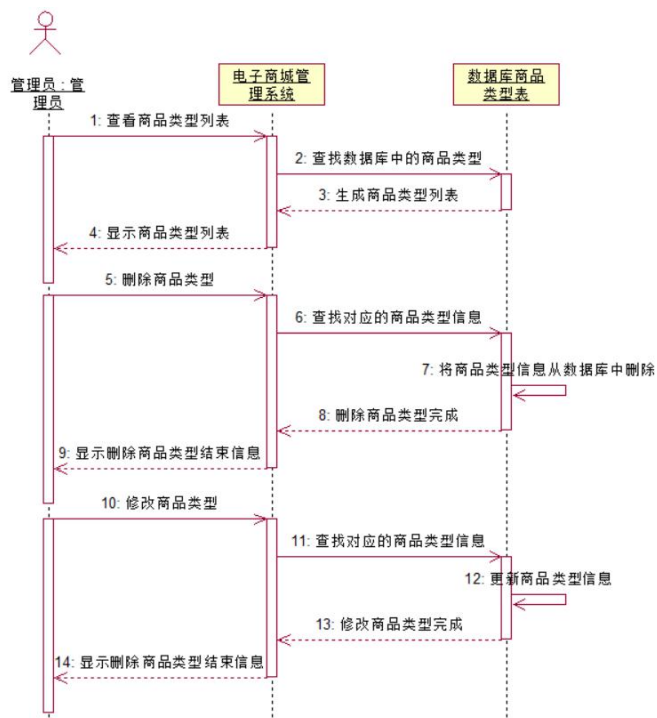
(2)管理员产品管理动态设计：

①管理员查看商品列表、查询/删除/修改商品



管理员可以使用商品管理功能，按类别或者默认查看商品列表，对个别具体商品的信息作出修改，或者删除某样商品，系统执行数据更新并返回处理结果。

③ 管理员查看商品类型列表、删除/修改商品类型

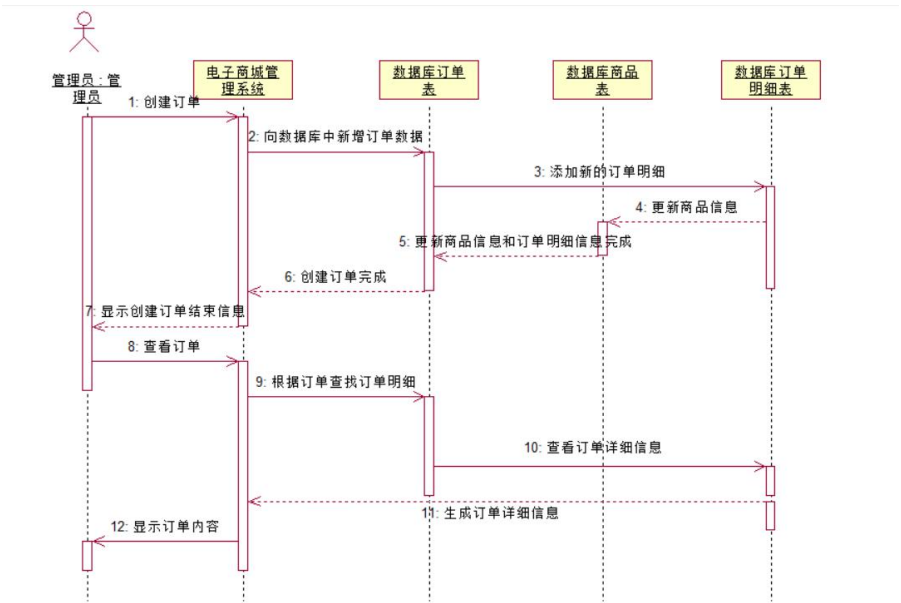


管理员可以使用商品类别管理功能，查看商品类别列表，对个别具体商品类型的信息作

出修改，或者删除某中商品类别，系统执行数据更新并返回处理结果。

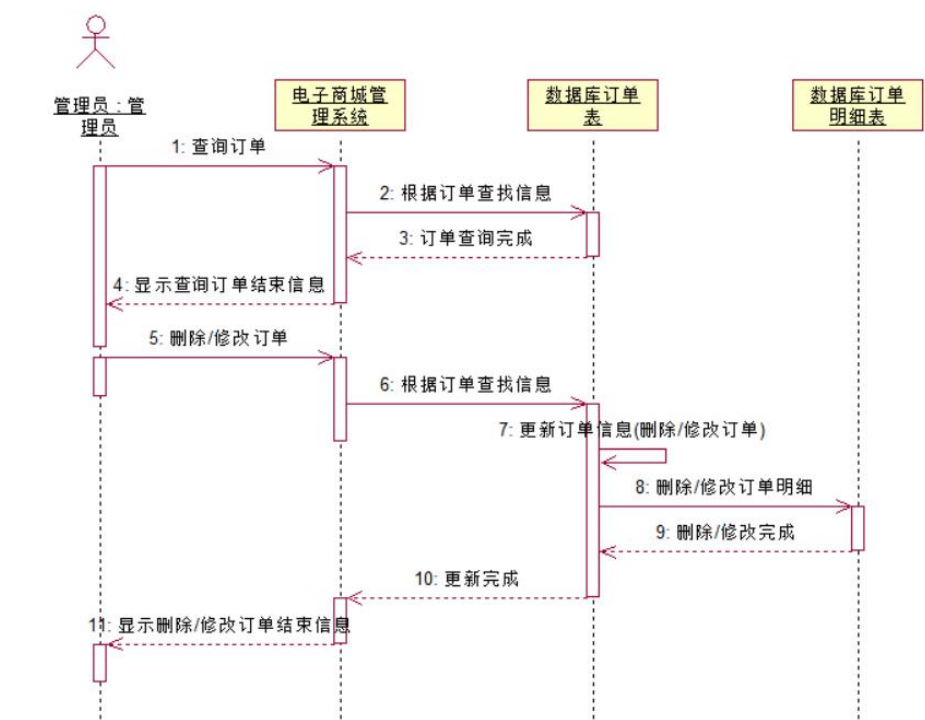
(3)管理员订单管理

①创建和查看订单



管理员在订单管理中创建新的订单信息，由系统录入数据库，和由用户生成的订单处理方法相同，更新商品库存等信息，之后返回执行结果。查看订单时，系统根据选择的订单查询内容，将具体内容显示在界面中。

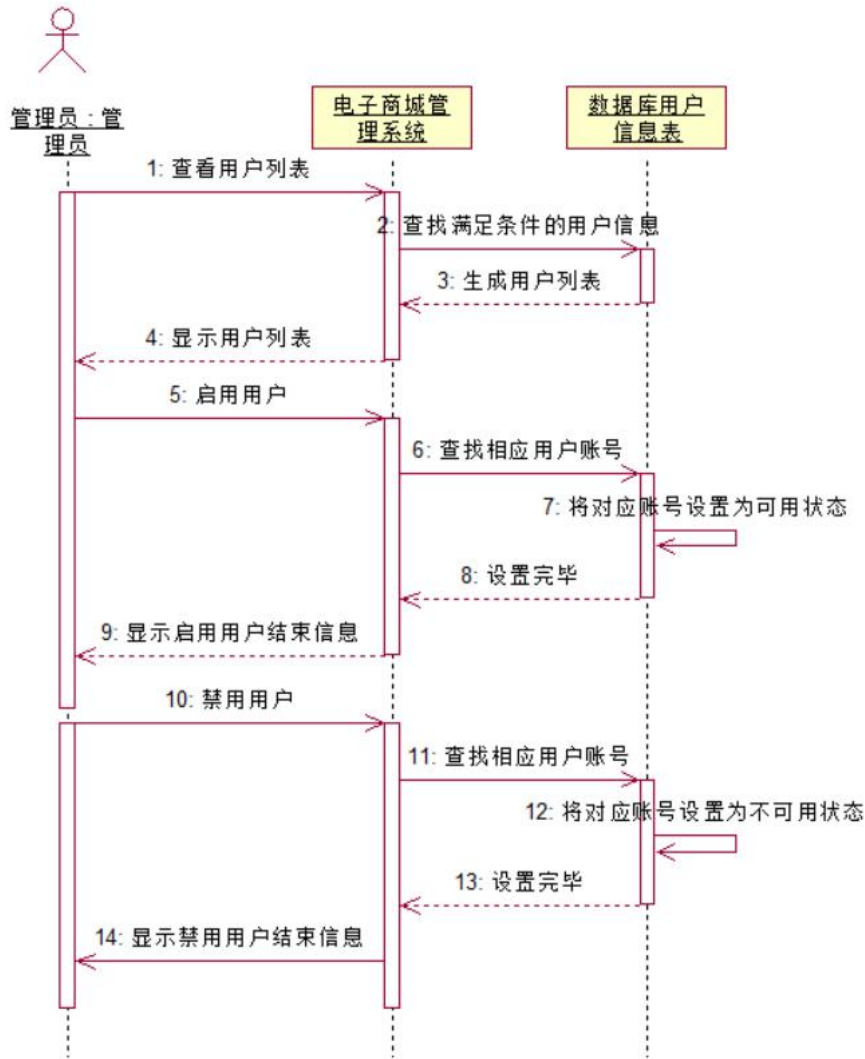
③ 查询、修改和删除订单



管理员在订单管理中提供订单的条件，查询订单信息，并可以选择对订单内容进行修改或修改，由系统更新数据之后返回执行结果。

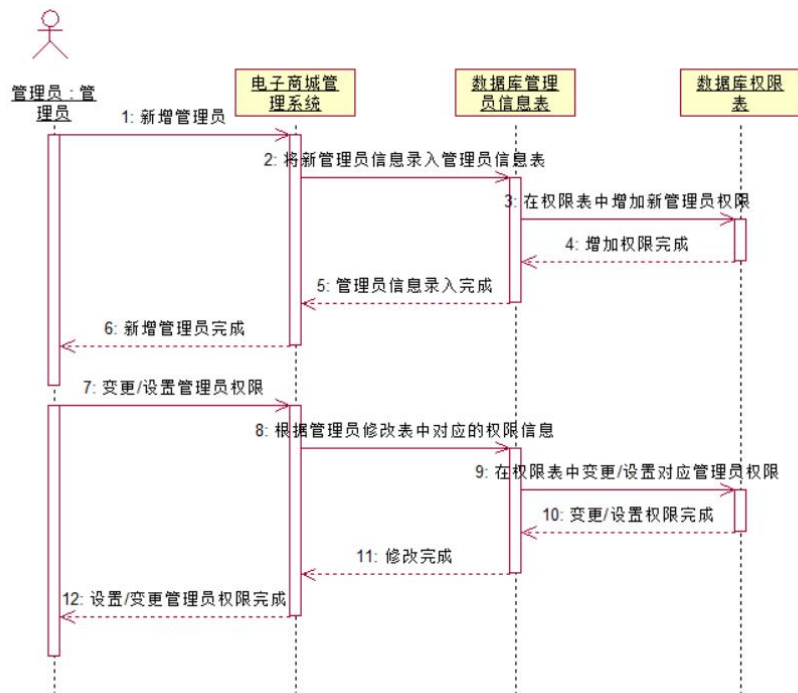
(4)管理员用户和权限管理

①查看用户列表、启用或禁用用户



②增加管理员、设置/变更管理员权限

管理员在用户管理中选择提供条件或默认查看用户的列表，系统查找所有满足条件的用户信息并显示在界面上。管理员启用/禁用用户时，系统更新对应用户状态为可用/不可用，限制用户对电子商城系统正常工能得使用，并返回更新结果。



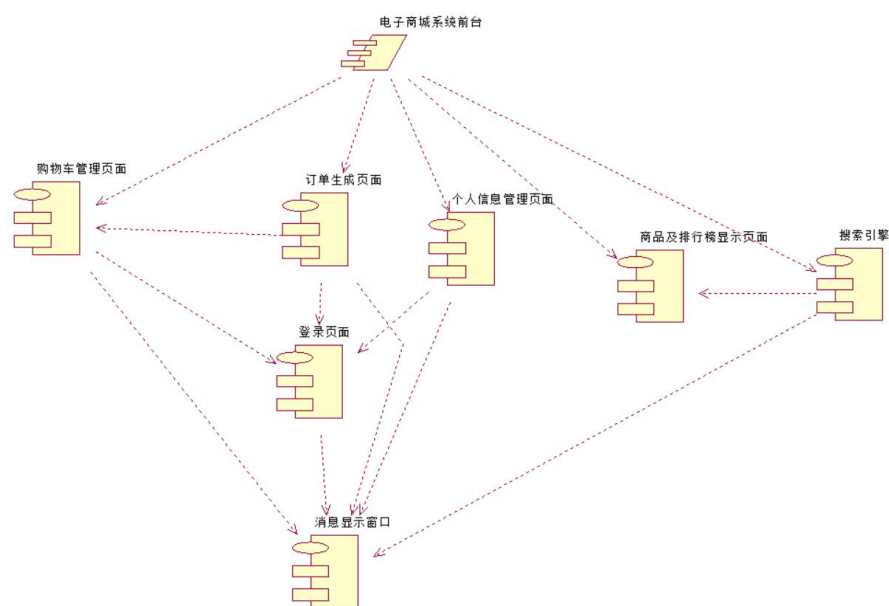
管理员在权限管理中选择新增管理员，系统将录入新增管理员信息并更新相关的权限数据，并返回执行结果到界面。设置管理员权限时，系统更新对应管理员的权限信息，并返回执行结果到界面。

五.系统实现模型设计

1.构件图

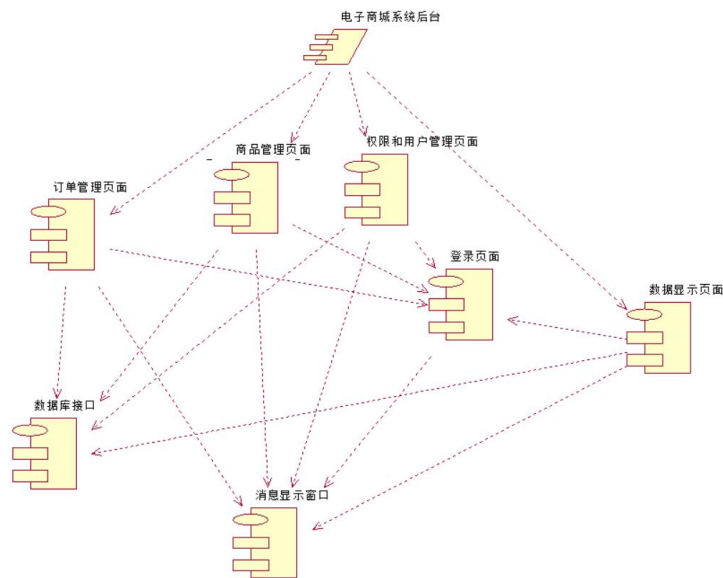
电子商城系统的构件图如下，整个网页系统分为前台和后台。

(1)前台系统的构件图



前台系统构件主要是维护各个页面的通信及显示的工具，保证数据在购物车管理页面、订单生成页面、登录页面、个人信息管理页面、商品及排行榜显示页面、搜索引擎等页面之间稳定有效的传输，并结合窗口工具显示反馈消息和网页的动态变化。

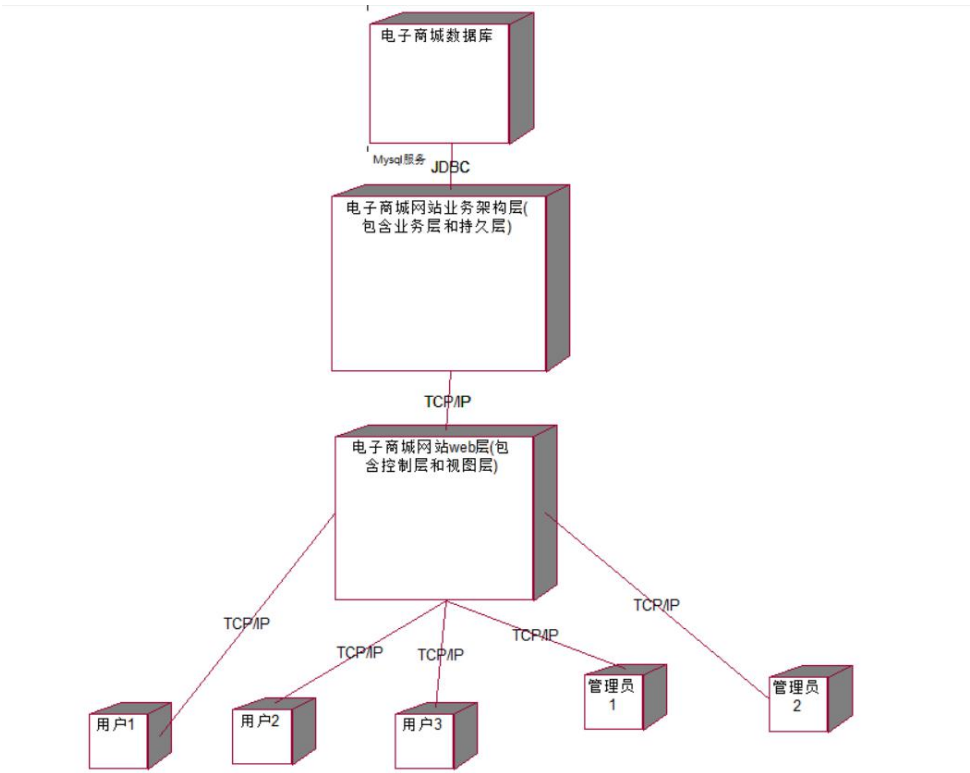
(2)后台系统的构件图



后台系统的构件除了维护各个页面的通信及显示的工具用以保证数据在订单管理页面、商品管理页面、权限和用户管理界面、数据显示页面、登录页面等页面之间稳定有效的传输和窗口工具显示反馈消息和动态变化之外，还包括与数据库的接口，建立起对数据库管理的通路。

2.配置图

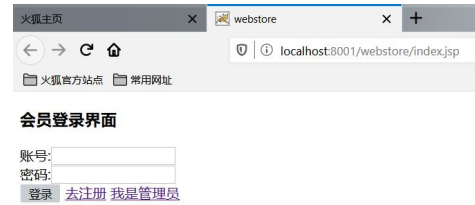
用配置图来反映电子商务系统硬件结构之上的软件结构及系统运行时的配置情况。电子商务系统的 web 应用保存在服务器中，服务器通过网络连接数据库管理数据，许多客户端通过网络连接服务器使用电子商务的功能。具体如下图所示：



六.用户注册和登录管理功能的实现

1.一般用户需要使用注册和登录功能

进入主页，显示会员登录界面：



会员登录界面

账号:

密码:

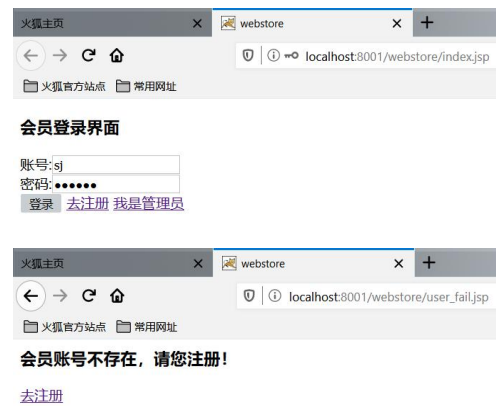
[登录](#) [去注册](#) [我是管理员](#)

此时已有用户信息表为空：

```
mysql> describe user_info;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| user_id | int | NO | PRI | NULL | auto_increment |
| user_name | varchar(16) | NO | | NULL | |
| user_password | varchar(16) | NO | | NULL | |
| realname | varchar(16) | YES | | NULL | |
| sex | varchar(4) | YES | | NULL | |
| address | varchar(255) | YES | | NULL | |
| question | varchar(50) | YES | | NULL | |
| answer | varchar(50) | YES | | NULL | |
| email | varchar(50) | YES | | NULL | |
| favorite | varchar(50) | YES | | NULL | |
| score | int | YES | | NULL | |
| regdate | datetime | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
12 rows in set (0.11 sec)

mysql> select * from user_info;
Empty set (0.00 sec)
```

输入账号密码，将无法匹配而登录失败。



会员登录界面

账号:sj

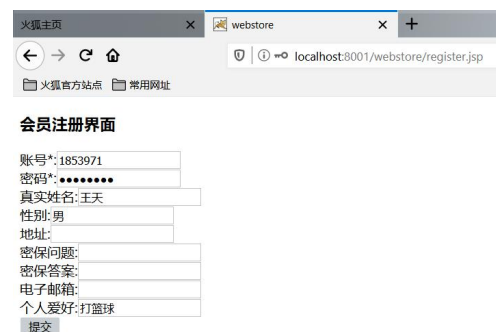
密码:*****

[登录](#) [去注册](#) [我是管理员](#)

会员账号不存在，请您注册!

[去注册](#)

因此需要注册会员账号，进入注册界面，填写个人信息(其中账号、密码为必填项，其余项可空着不填)：



会员注册界面

账号*: 1853971

密码*: *****

真实姓名: 王天

性别: 男

地址:

密保问题:

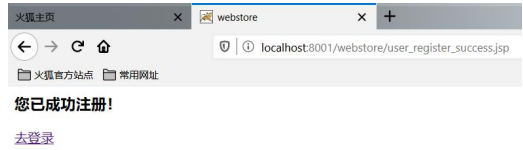
密保答案:

电子邮箱:

个人爱好: 打篮球

[提交](#)

提交个人信息后显示注册成功：



此时数据库用户表中就会添加新注册的会员信息：

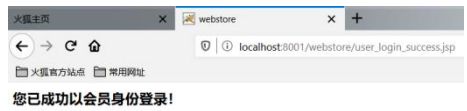
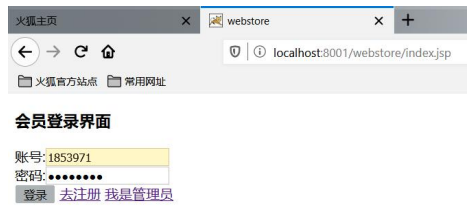
```
mysql> describe user_info;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| user_id | int | NO | PRI | NULL | auto_increment |
| user_name | varchar(16) | NO | | NULL | |
| user_password | varchar(16) | NO | | NULL | |
| realname | varchar(16) | YES | | NULL | |
| sex | varchar(4) | YES | | NULL | |
| address | varchar(255) | YES | | NULL | |
| question | varchar(60) | YES | | NULL | |
| answer | varchar(50) | YES | | NULL | |
| email | varchar(50) | YES | | NULL | |
| favorite | varchar(50) | YES | | NULL | |
| score | int | YES | | NULL | |
| regdate | datetime | YES | | NULL | |
+-----+-----+-----+-----+-----+-----+
12 rows in set (0.11 sec)

mysql> select * from user_info;
Empty set (0.00 sec)

mysql> select * from user_info;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| user_id | user_name | user_password | realname | sex | address | question | answer | email | favorite | score | regdate |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | 1853971 | wangtian | 王天 | 男 | NULL | NULL | NULL | NULL | 打篮球 | 0 | 2020-06-01 21:51:35 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

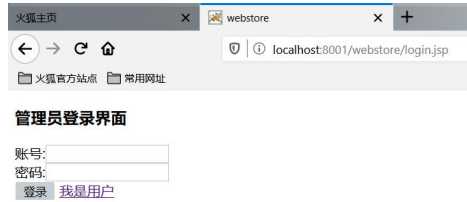
mysql>
```

此时使用新注册的账号即可完成会员的登录：

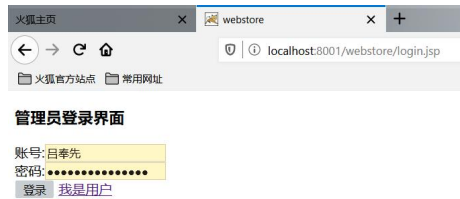


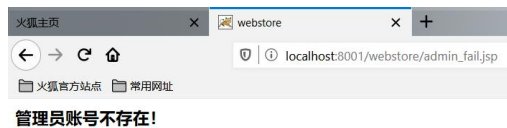
2. 管理员需要使用登录功能

从用户登录界面可以跳转到管理员登录界面：



若输入的管理员账号不存在则会出现提示：





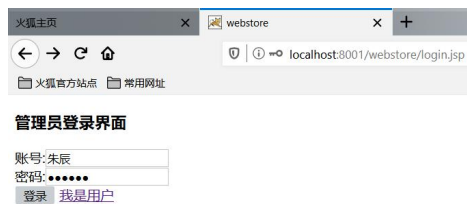
管理员并非通过注册生成，而是更高权限的管理员设置新管理员，因此在数据库 `admin_info` 表中先添加管理员：

```
mysql> describe admin_info;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| admin_id   | int           | NO   | PRI | NULL    | auto_increment |
| admin_name | varchar(16)   | YES  |     | NULL    |                |
| pwd        | varchar(16)   | YES  |     | NULL    |                |
| role       | varchar(8)    | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> insert into admin_info (admin_name,pwd) values (<"朱辰","wowcyh");
Query OK, 1 row affected (0.45 sec)

mysql> select * from admin_info;
+-----+-----+-----+-----+
| admin_id | admin_name | pwd   | role |
+-----+-----+-----+-----+
| 1        | 朱辰      | wowcyh | NULL |
+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

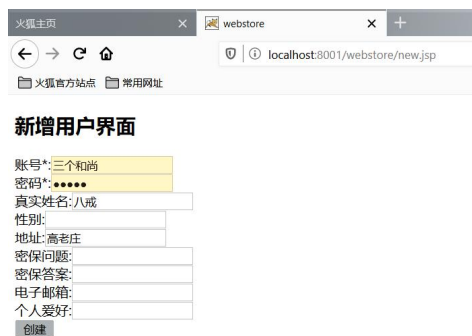
之后使用正确的管理员账号即可登录成功：



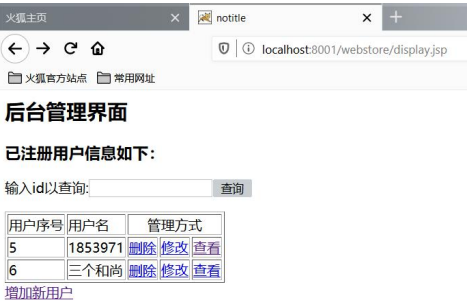
管理员登录后到达后台管理界面，可以看到已经注册的用户的列表，并且可以执行相应的管理功能。



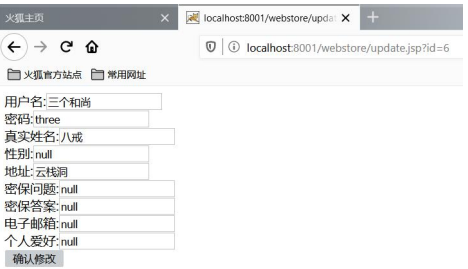
可以执行增加新用户的功能：



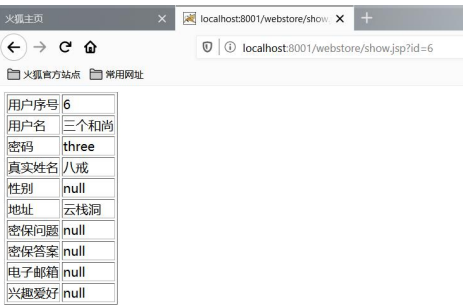
点击创建后即创建完毕。后台的用户列表就会发生变化。



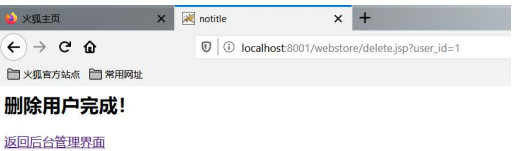
可以修改用户的信息：



点击确认修改即可完成用户信息的修改。
之后使用查看用户的功能可以看到用户的详细信息。



可以将用户删除：



再到用户列表可以观察到用户数量减少了：



在查询框输入用户 id 可以查找相应用户：

火狐主页

notitle

+

←

→

↺

🏠

localhost:8001/webstore/display2.jsp?id=5

📁 火狐官方网站

📁 常用网址

查询界面

查询到的用户如下:

用户序号	用户名	管理方式
5	1853971	删除 修改 查看

如果没有满足 id 条件的用户则会出现提示:

火狐主页

webstore

+

←

→

↺

🏠

localhost:8001/webstore/find_fail.jsp

📁 火狐官方网站

📁 常用网址

未查询到结果!

[返回](#)

从用户列表界面可以切换到管理员管理界面:

火狐主页

notitle

+

←

→

↺

🏠

localhost:8001/webstore/display3.jsp

📁 火狐官方网站

📁 常用网址

后台管理界面

系统已有管理员信息如下:

输入id以查询:

查询

管理员序号	管理员名	管理方式
1	朱辰	设置 查看
2	12345	设置 查看

[增加管理员](#) [管理用户](#)

可以增加管理员:

火狐主页

webstore

+

←

→

↺

🏠

localhost:8001/webstore/new2.jsp

📁 火狐官方网站

📁 常用网址

新增管理员界面

账号*:朱嘉

密码*:.....

权限:一般

创建

火狐主页

webstore

+

←

→

↺

🏠

localhost:8001/webstore/add_success.jsp

📁 火狐官方网站

📁 常用网址

增加管理员成功

[返回](#)

此时如果新增的管理员名称已经存在则会进行提示:

火狐主页

webstore

+

←

→

↺

🏠

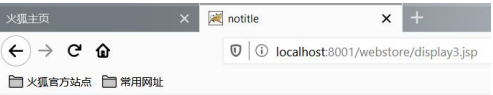
localhost:8001/webstore/add_fail.jsp

📁 火狐官方网站

📁 常用网址

管理员已存在!

[返回](#)



后台管理界面

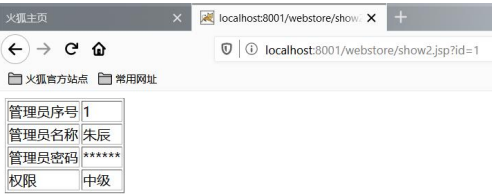
系统已有管理员信息如下:

输入id以查询:

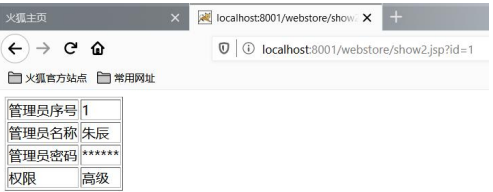
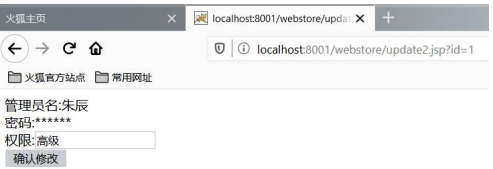
管理员序号	管理员名	管理方式
1	朱辰	设置 查看
2	12345	设置 查看
3	朱燕	设置 查看

[增加管理员](#) [管理用户](#)

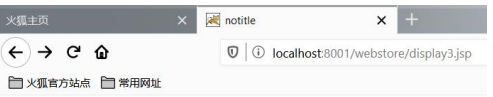
可以查看管理员的权限:



也可以通过设置对管理员权限进行修改:



在管理员列表可以输入 id 以查询对应管理员:



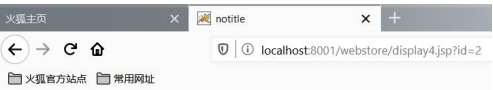
后台管理界面

系统已有管理员信息如下:

输入id以查询: 2

管理员序号	管理员名	管理方式
1	朱辰	设置 查看
2	12345	设置 查看
3	朱燕	设置 查看

[增加管理员](#) [管理用户](#)

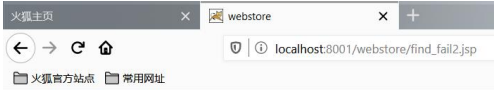


查询界面

查询到的管理员如下:

管理员序号	管理员名称	管理方式
2	12345	设置 查看

不存在对应的管理员则会进行提示：



未查询到结果!

[返回](#)