

1 引言	2
1.1 编写目的	2
1.2 背景	2
1.3 定义	2
1.4 参考资料	2
2 系统结构模型	3
2.1 初始对象表	3
2.2 初始类图	4
2.3 细化类图	4
2.3.1 属性的识别	5
2.3.2 操作的识别	8
2.4 三层精化类图	10
3 系统行为模型	13
3.1 交互模型	13
3.2 状态模型	17

概要设计说明书

(本报告工作量袁沛霖 > 李达)

1 引言

1.1 编写目的

编写本说明书的目的是说明系统的静态结构及动态结构,包括初始的类模型、细化的类、动态交互模型,为进一步的设计和开发提供依据。

预期读者为:软件开发的人员,项目评审人员,及软件测试人员。

1.2 背景

说明:

1. 待开发的软件系统的名称: 基于 Android 的线下交易购物平台;
2. 本项目的任务提出者: 袁沛霖, 李达;
3. 本项目的任务开发者: 袁沛霖, 李达;
4. 用户及实现该软件的计算中心或计算机网络:;
5. 该软件系统同其他系统或其他机构的基本的相互来往关系:。

1.3 定义

1. 用户输入: 用户输入的用户名及密码;
2. 用户名及口令与用户分类信息: 存放在一个数据库表中,用以判断该用户是商家用户还是买家用户;
3. 买家用户信息管理: 是商家用户使用该系统的权限的表现,只能处理自己的信息;
4. 商家用户信息管理: 即档案信息,存放商家的名称,地理位置,展示的商品等各项基本信息,买家可以查询,商家自身可以对状态进行发布,修改,删除;
5. 出错显示: 当用户名与密码不正确时,系统显示出错,并要求用户重新输入;
6. 显示屏: 是用户与软件系统进行交互的显示渠道。

1.4 参考资料

1. 孙涌等编,现代软件工程,北京希望电子出版社 2002 年
2. 齐治昌等,软件工程(第二版),高等教育出版社,2004
3. Pressman R S. Software Engineering: A Practitioner's Approach. 3rd
4. 郑人杰等,实用软件工程(第二版),清华大学出版社,1997 任务概述

2 系统结构模型

2.1 初始对象表

通过对需求的分析，提炼出可能的对象/类，这是面向对象系统分析设计的关键，也是难点所在。为了准确识别系统的类和对象，从以下三个方面进行考虑：

1、考虑系统责任

对照系统责任所要求的每一项功能，查看是否可以由现有的对象完成这些功能。如果发现某些功能在现有的任何对象中都不能提供，则可启发我们发现领域某些遗漏的对象。

2、名词技术

从名词到对象或类通常有一对一的映射。运用名词、代词和名词短语识别对象和类。可以启发发现对象的因素包括：人员、组织、物品、设备、事件（如索赔、上访、交易）、表格、日志、报告、结构（汽车上面有车辆，下面有客车和轿车，左右有摩托车；内有发动机）等。

3、从用例中识别

- （1）用例描述中出现哪些实体？需要哪些实体的合作？
- （2）用例执行过程中会产生并存储哪些信息？
- （3）用例要求与之相关的每个角色的输入是什么？
- （4）用例反馈与之关联的每个角色的输出是什么？
- （5）用例需要操作哪些设备？

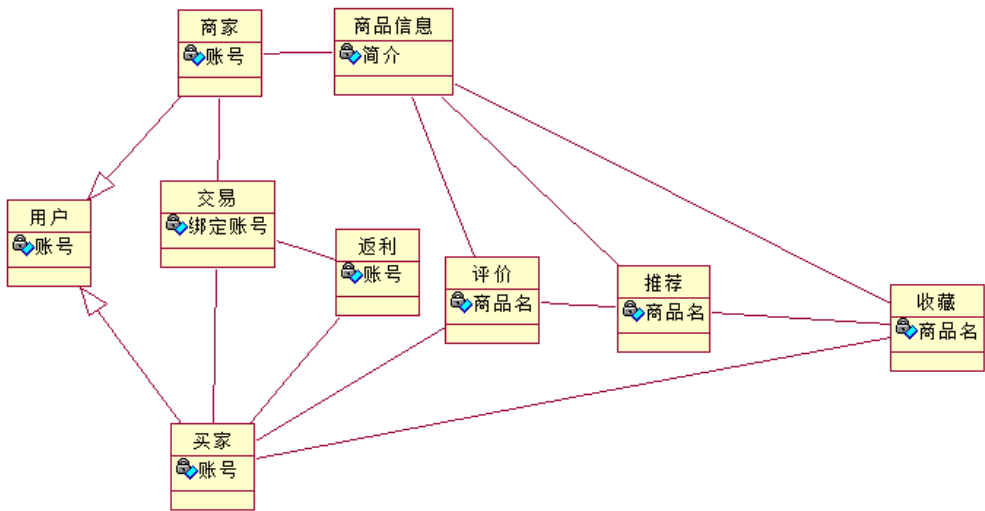
通过对线下购物交易平台系统的分析，建立初始对象/类表如下：

可能的类	英文类名	简单描述
用户	User	用户分两类：买家与商家，存储 ID、密码等基本信息
商品信息	Store	存储商品的基本信息如商品描述、定价等，可进行发布、修改、删除等操作
推荐	Recommend	收集 Evaluation 类中商铺好评率和 Favorite 类中的信息进行

		分析
收藏	Favorite	用来存储喜爱的 StoreInformation 类中的对象
评价	Evaluation	用来实现对商铺的评价功能
交易	Trade	用来实现借助第三方支付平台交易功能
返利	Reward	用来实现每天结算的类似于中奖的返利功能

2.2 初始类图

这一部的分析重点在于类和类间关系的表示,即系统体系结构的建立,因此对类的描述,只要给出其关键字即可,属性和操作的细化工作可以在类模型完成后再进行。通过分析,线下一交易购物平台的初始类图如图所示。



把箭头去掉吧，非单向

2.3 细化类图

初始类图奠定了系统的体系结构,对象/类还需进一步细化描述其属性和操作,为下一步数据库设计和模块实现做准备。首先附上需求报告的用例图如下:这是活动图啊,呆

2.3.1 属性的识别

确定属性可以通过提出以下问题得到：

- ❑ 按常识这个对象应该有哪些属性？
- ❑ 在当前的问题域中，对象应该有哪些属性？
- ❑ 根据系统责任，这个对象应具有哪些属性？
- ❑ 建立这个对象是为了保存和管理哪些信息？
- ❑ 对象为了完成其功能，需要增设哪些属性？
- ❑ 对象是否需要通过专设的属性区别其状态？
- ❑ 用什么属性表示聚集和关联？
- ❑ 可利用需求文档中的形容词或所有格短语。

类名：用户

属性表如下表所示：

中文名	英文名	数据类型
账号	userID	String
密码	password	String
权限	power	Int
绑定支付账号	tradeID	String

类名：商品信息 ~~到底是商铺还是商品呢？~~

属性表如下表所示：看属性表，是商品，类图里没有商品呀

中文名	英文名	数据类型
商品号	goodsID	String
商品名	goodsName	String
商品价格	goodsPrice	Float
商品简介	goodsInformation	String

地址	storeAddress	String
联系方式	storePhone	String

类名：推荐 纯计算类，无需存推荐结果，无自有属性

类名：收藏

属性表如下表所示：

中文名	英文名	数据类型
收藏者	userID	String
收藏时间	favoriteTime	String
收藏商品名	favoriteID	String
收藏商铺链接	favoriteLink	String

类名：评价

属性表如下表所示：

中文名	英文名	数据类型
评价 ID	evaluationNumber	String
评价者	userID	String
评价时间	evaluationTime	String
评价对象名 对商品评价还是对整个订单？(单个商品)	goodsID	String
评价内容	evaluationInformatin	String
评价等级	evaluationGrade	Int

类名：交易

属性表如下表所示：

中文名	英文名	数据类型
收款方账号	sellID	String
付款方账号	buyID	String

系统账号 这是啥？(这是在能调用第三方支付接口的情况用以存储 1%交易额和返利的账号)	systemID	String
交易状态	tradeStatus	boolean

订单

消费者	buyer	String
收费者	seller	String
交易号	tradeOrder	Int
交易时间	tradeTime	String

交易记录

交易号	tradeOrder	Int
交易商品名	tradeID	String
物品号	goodsNumber	Int
交易价格	tradePrice	Float
交易数量	tradeAmount	Int

类名：返利（在能调用第三方支付接口的情况下予以实现）

属性表如下表所示：

中文名	英文名	数据类型
获利者 ID	rewardID	String
获利者绑定账号	UserID	String
谁返的（由系统账户进行返利操作）		
实时返利金额	rewardNumber	Float
时间	rewardTime	String

2.3.2 操作的识别

1、操作的分类

基本操作：包括数据库检索和更新，如增加、删除、修改、分类、选择、查询、计算、汇总

关键操作：必须由对象提供的、在算法上复杂的业务操作（如要进行某些计算或监控操作）。

2、操作的识别

确定属性可以通过提出以下问题得到：

- 有哪些类会与该类交互？
- 所有与该类具有交互行为的类会发送哪些消息给该类？该类又会发送哪些消息给这些类？
- 该类如何响应别的类发送来的消息？在发送消息之前，该类需要做何处理？
- 从该类本身来说，它应该具有哪些操作来维持其信息的更新、一致性和完整性？
- 系统是否需要该类有另外一些职责？

类名：用户

操作表如下表所示：

中文名	英文名	返回值类型
增加一个用户	addUser()	void
删除一个用户	deleteUser()	void
修改用户信息	updateUser()	void
判断是否合法用户	isUser()	boolean
判断用户权限	getPower()	Int
查询用户信息	getUsers()	User
绑定支付账号	addPay()	void
取消绑定	cancel()	void

类名：商铺信息

操作表如下表所示：

中文名	英文名	返回值类型
判断是否商家	isSeller()	boolean
显示目前商品	goodsView()	void
新增商品	addGoods()	void
修改商品信息	updateGoods()	void
删除商品信息	deleteGoods()	void
查询商品信息	queryGoods()	void

类名：推荐

操作表如下表所示：

中文名	英文名	返回值类型
获取好评率信息	getEvaluationInformation())	Int
获取购买商品名	getTradeName()	String
获取收藏记录	getFavoriteInformation()	Favorite
推荐计算	recommendInformation()	Store

类名：收藏

操作表如下表所示：

中文名	英文名	返回值类型
添加收藏	addFavorite()	void
删除收藏	deleteFavorite()	void
确认是否收藏	ensureFavorite()	boolean
确认是否取消收藏	ensureDelFavorite()	boolean

类名：评价

操作表如下表所示：

中文名	英文名	返回值类型
-----	-----	-------

添加评价	addEvaluation()	void
删除评价	deleteEvaluation()	void
判断是否已评价	isEvaluated()	boolean

类名：交易

操作表如下表所示：

中文名	英文名	返回值类型
获取买家绑定账号	getBuyerID()	String
获取卖家绑定账号	getSellerID()	String
交易	pay()	void
生成交易信息	generateTradeInformation()	String

类名：返利

操作表如下表所示：

中文名	英文名	返回值类型
显示返利金额	showReward()	void
获取返利买家	getLuckyBuyer()	User
返利	rewarding()	void

2.4 三层精化类图

在初始类图的基础上，运用三层体系结构思想对基本类模型进行划分。根据各个对象的具体情况，将对象分成界面、事务和数据层。

1、对用户类的三层划分

界面层： login.jsp 登录界面

login_failure.jsp 登录失败界面，提示错误所在

user.jsp 用户登录成功返回的界面

register.jsp 注册界面

registerfailure.jsp 注册失败界面

registersuccess.jsp 注册成功界面

事务层: login.java 登录处理

Register.java 注册处理

数据层: user.dat 用户信息表

2、对商品信息类的三层划分

界面层: goods_add.jsp 增加商品界面

goods_possess.jsp 出售的商品界面

goods_update.jsp 修改商品信息界面

goods_query.jsp 商品查询界面

goods_single.jsp 单个商品信息显示界面

goods_list.jsp 商品信息列表显示界面

事务层: store.java 商铺信息管理类

数据层: user.dat 用户信息表

store.dat 商品信息表

3、对推荐类的三层划分

界面层: recommend.jsp 推荐商品列表界面

事务层: recommend.java 推荐算法类

数据层: store.dat 商品信息表

favorite.dat 收藏信息表

evaluation.dat 评价信息表

4、对收藏类的三层划分

界面层: favorite_list.jsp 收藏商铺列表界面

favorite_ensure.jsp 确认收藏界面

favorite_delete_ensure.jsp 确认删除收藏界面

事务层: favorite.java 收藏信息管理类

数据层: favorite.dat 收藏信息表

store.dat 商品信息表

5、对评价类的三层划分

界面层: evaluation_add.jsp 增加评价界面

evaluation_delete_ensure.jsp 确认删除评价界面

事务层: evaluation.java 评价操作管理类

数据层: evaluation.dat 评价信息表

6、对交易类的三层划分

界面层: trade.jsp 交易主界面

generate_code.jsp 二维码生成界面

scan_code.jsp 二维码扫描界面

trade_ensure.jsp 交易确认界面

事务层: trade.java 交易处理类

数据层: trade.dat 交易信息表

7、对返利类的三层划分

界面层: reward.jsp 返利显示界面, 包括实时金额与前日幸运买家

事务层: reward.java 返利处理类

数据层: reward.dat 返利信息表

user.dat 用户信息表

注意, 除了类图中的类以外, 系统类的提炼也非常关键。如有必要提取系统的数据库连接类, 负责完成所有的数据库连接, 该类没有界面层, 没有数据层, 只有事务层 dbConnector.java。

3 系统行为模型

3.1 交互模型

交互是指共同完成特定任务的一组对象之间交换消息的行为，用于刻画系统的控制流，存在于对象协作或操作实现过程中。

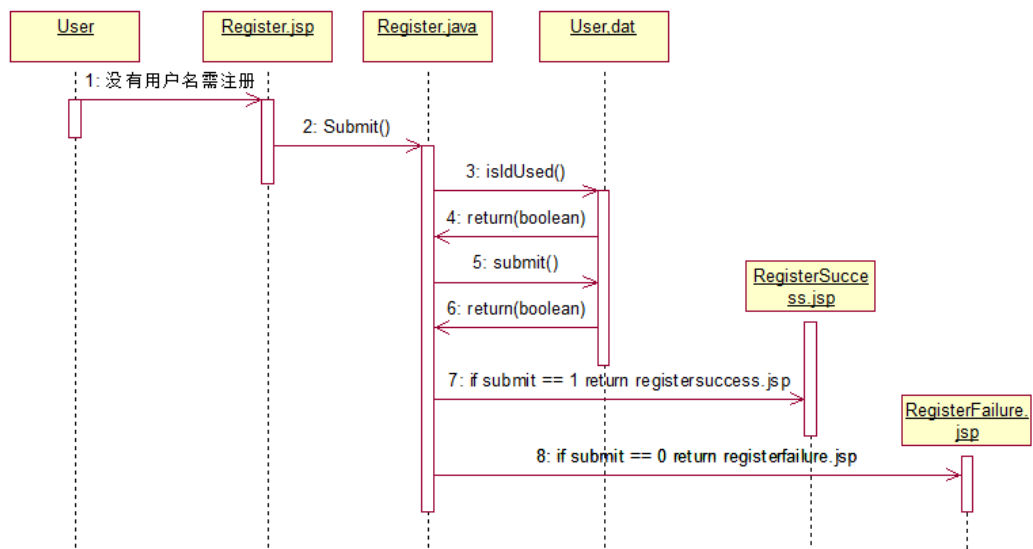
交互模型包括顺序图和协作图，顺序图强调时间顺序，协作图强调对象间的合作关系，顺序图与协作图可相互转换，无信息损失。顺序图和合作图的选择原则如下：

- （1）如果对象数目不多，交互情况不复杂，应选择顺序图；
- （2）如果系统关系对象交互行为的时间特性，应选择顺序图；
- （3）如果对象数目很多，且交互情况较复杂，可能使用协作图，但其中的某些“场景片段”可以使用顺序图来专门描述其时间特性。

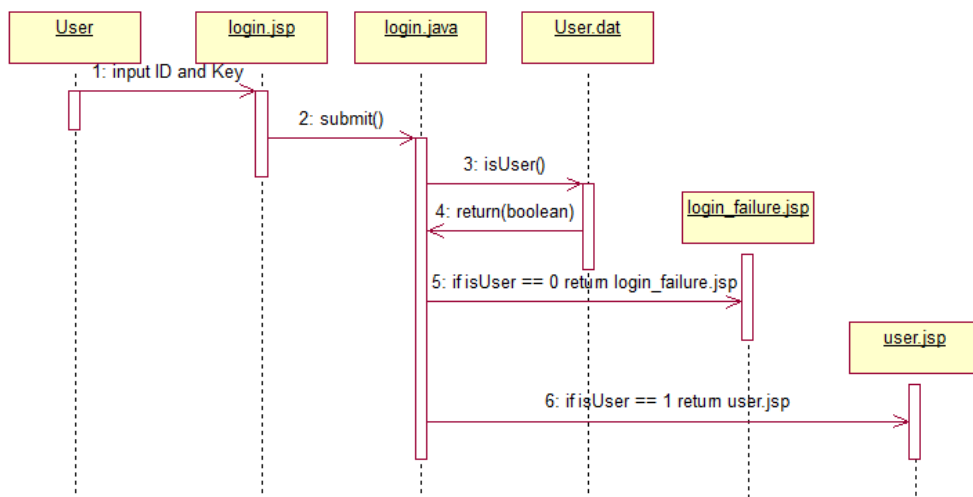
交互模型的建模步骤：

- （1）列出用例相关的所有对象（类）；
- （2）根据活动及类间关系确定对象间的消息通讯；
- （3）定义对象间的消息连接和消息格式；
- （4）确定消息发生的时间顺序；
- （5）确定消息编号（针对协作图）；
- （6）画出交互模型。

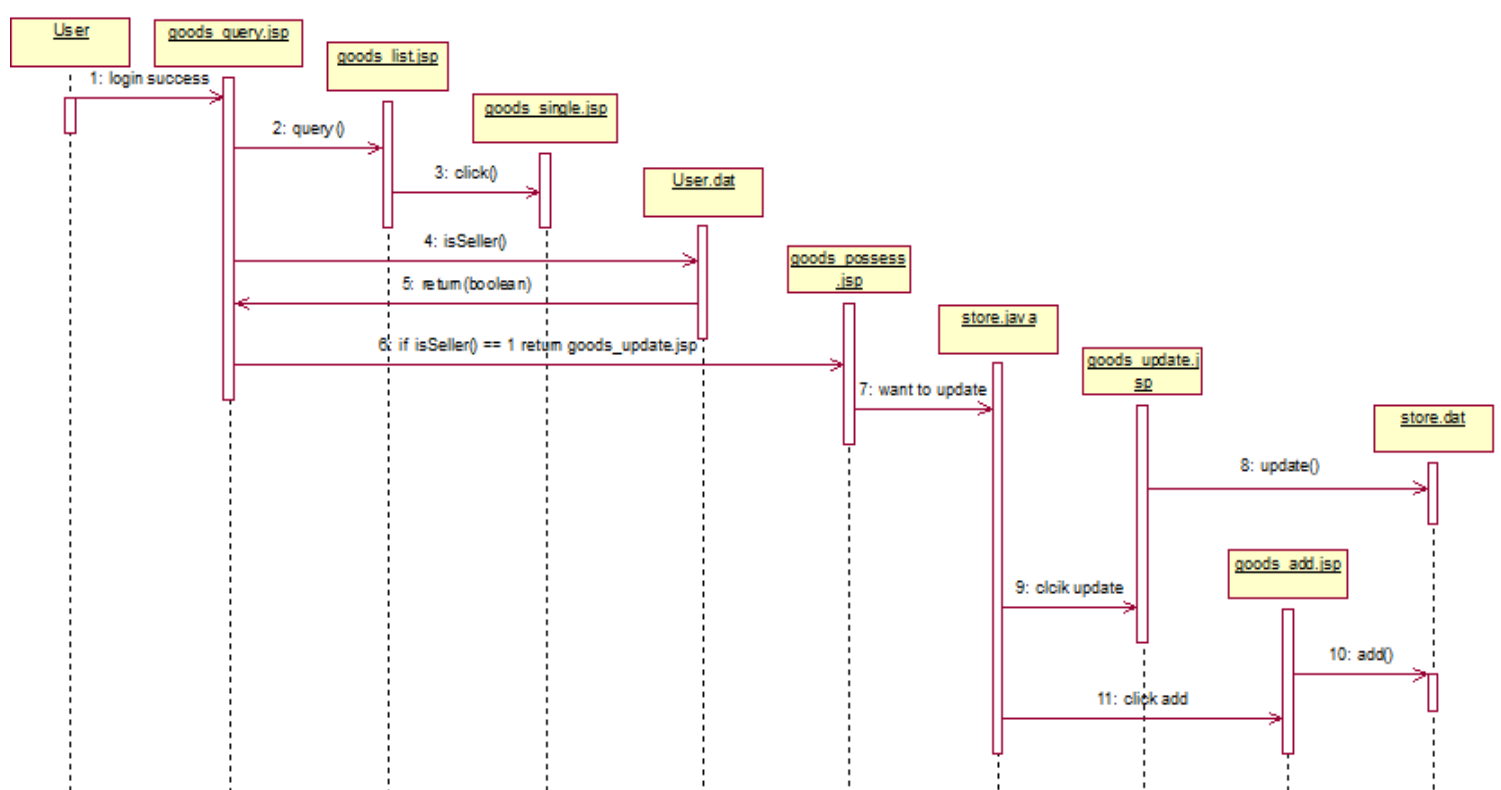
1、注册顺序图



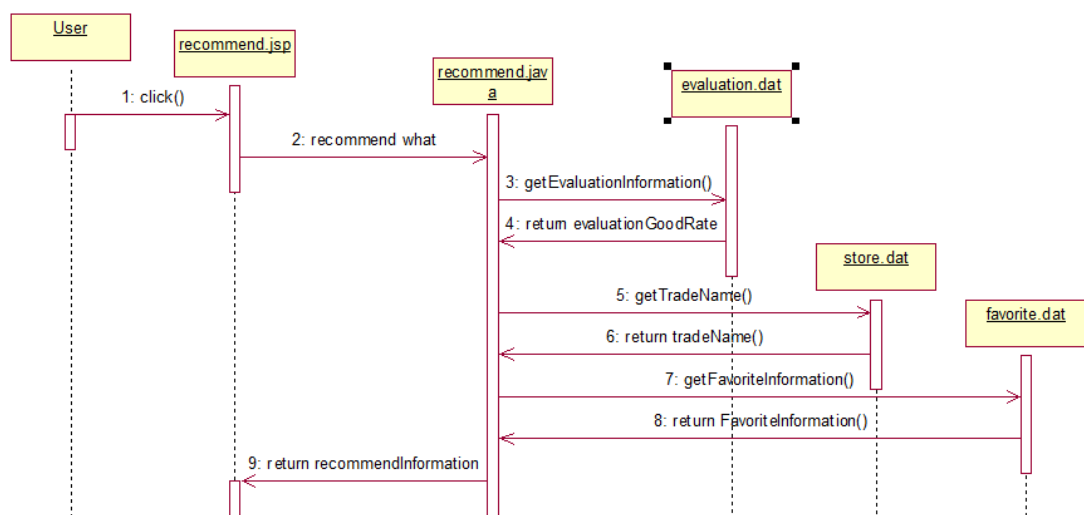
2、登陆顺序图



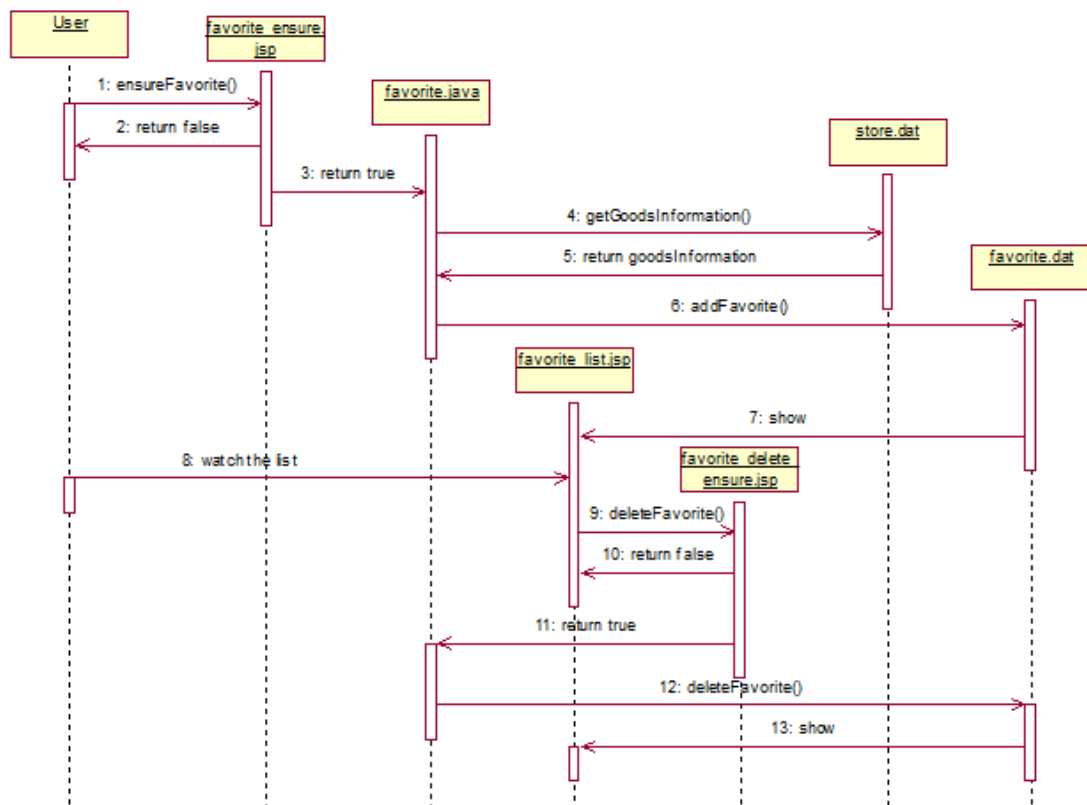
3、商品信息管理顺序图



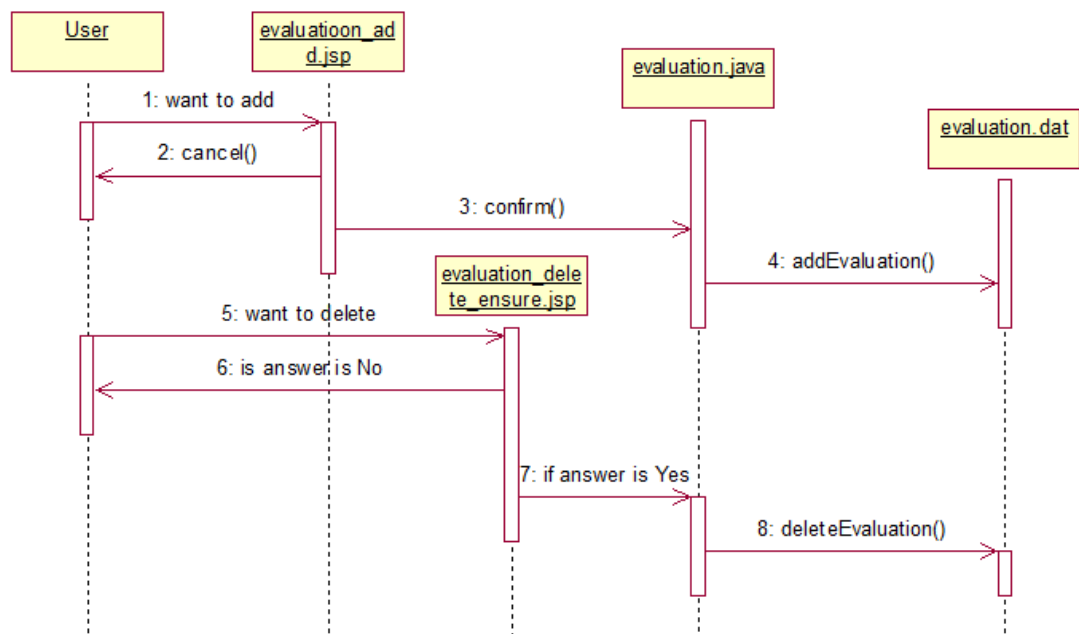
4、推荐顺序图



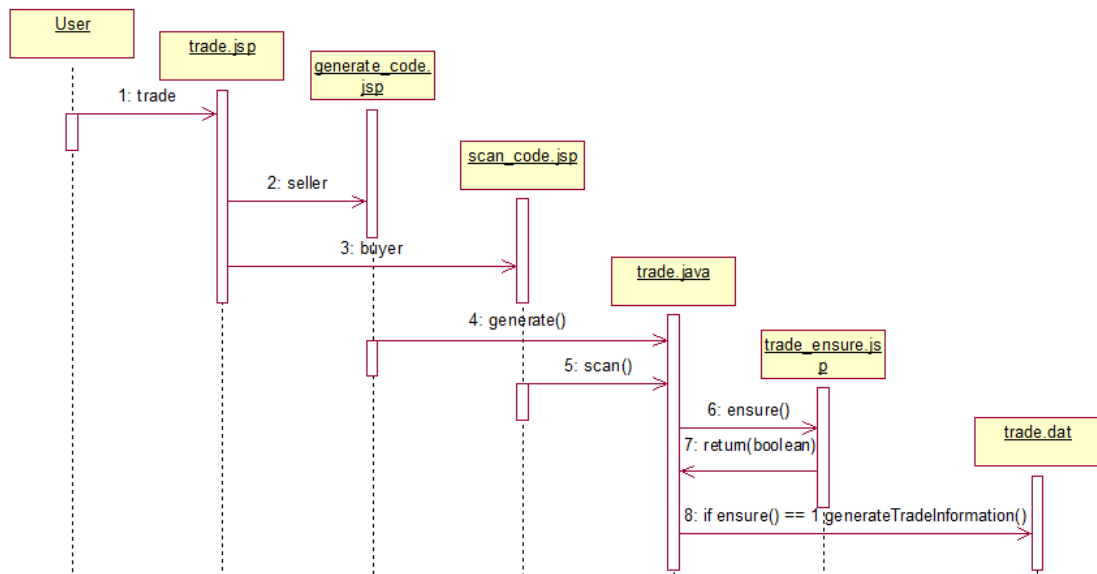
5、收藏顺序图



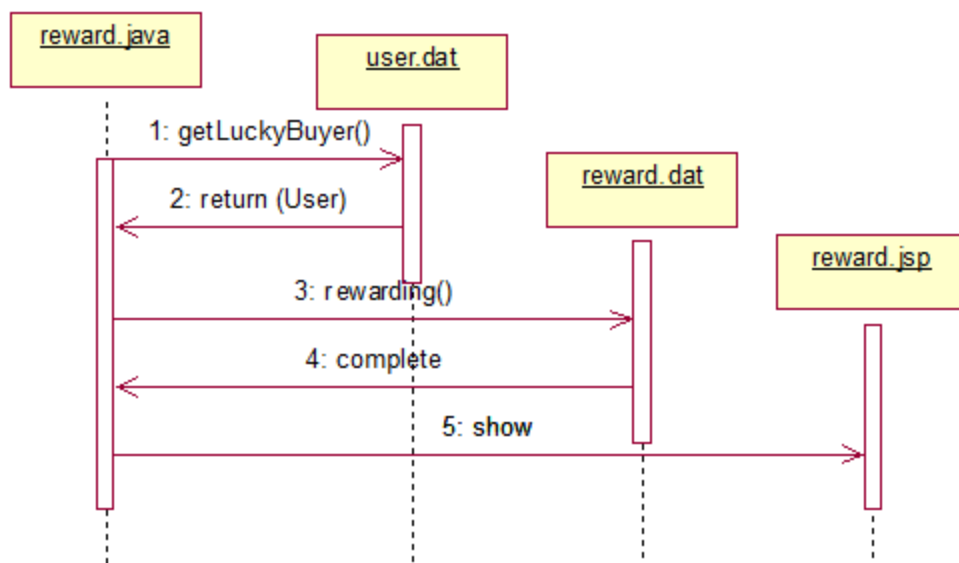
6、评价顺序图



7、交易顺序图



8、返利顺序图



3.2 状态模型

状态模型用于描述单个对象在生命期内响应事件所经历的状态及转移行为，是由状态及活动、转换及事件脚本组成的状态机。一个状态机可表示成一个状态图，体现从状态到状态的控制流。

并不需要为系统的每个对象建立状态图，在大多数情况下，仅对这样的对象的建立状态

图：它具有清晰的生命期，对其语境外部的事件能做出反应，且它的当前行为受它的过去行为的影响，这样的对象一般为被动型对象。

注意：不是所有对象都能用状态图来描述，如果一个对象的属性列表里有专门用来描述器状态的属性，且该属性的取值是枚举型，这样的对象需要用状态图表示。一般来说，这种对象所处的状态对系统操作有重大影响，要通过判断这个状态取值来决定能做什么操作。

交易对象的状态变化

