详细设计

**修订历史**

|  |  |  |  |
| --- | --- | --- | --- |
| **版 本** | **日 期** | **描 述** | **作 者** |
| **v1.0** | **2018.10.22** | 完成概述部分 | **林少辉** |
| **V1.1** | **2018.10.24** | 系统复用计划 | **刘斌** |
| **v1.2** | **2018.10.26** | 系统内外部接口设计 | **刘斌** |
| **V1.3** | **2018.10.26** | 对象模型设计 | **游增** |
| **V1.4** | **2018.10.30** | 系统顺序图 | **林水泉** |
| **v1.5** | **2018.10.30** | 系统状态图 | **汪喆** |
| **V1.6** | **2018.10.29** | 非功能性需求 | **林烁** |
| **v1.6** | **2018.10.31** | 根据各部分，补充术语表 | **林少辉** |
| **v2.0** | **2018.10.31** | 小组讨论，对概要设计进行整体修订 | **全体成员** |

## 概述

1.1 系统概述

SCUT Mall是一个模仿京东的购物网站，分为客户端和管理员端。客户端主要用于让用户进行购物、操作购物车、查询商品信息、打折促销活动以及用户购物统计功能；而管理员端用于系统管理员、业务员进行商品分类管理和销售数据统计等场景。

1.2软件设计目标

其软件设计目标是实现一个采用JavaEE开发、包括账户模块、商品管理模块、查询模块、购物模块、系统管理模块、统计模块等六大模块的Java Web网页。其中账户模块是许多IT系统的基本模块，用户可以注册、登录账户，并编辑修改账户的身份、地址信息。商品管理模块是SCUT网上购物系统的核心模块之一，商品管理是指商品上推到系统中，呈现在用户的可浏览视图中以供购买，商品下架则是不开放该商品的购买权限以及移出用户的可浏览识图。查询模块是SCUT网上购物系统的核心模块之一，用户可以查询在售商品（即已上架的商品）、查询购物车中的商品和查询已购买的商品。购物模块是SCUT网上购物系统的核心模块之一，用户对心仪的在售商品或购物车中的商品进行购买产生结算，主要有订单结算、积分结算和打折结算。系统管理模块是SCUT网上购物系统的支撑模块之一，管理员通过管理界面查看用户信息、业务员信息、商品信息、业务员/统计人员授权、用户封号等操作，以及开启和关闭系统。统计模块是SCUT网上购物系统的核心模块之一，统计人员根据系统的业务数据进行相关的用户、订单、营业额的统计，为商业决策提供建议。

而在具体的开发上，我们选择了springboot框架，使得开发更加容易入手；而MVC模式的使用，有利于将代码分解，逻辑更加清晰；此外，我们还将使用Tomcat作为服务器，租用阿里云的入门级服务器，这些资源都是不可或缺的！

在非功能性需求方面，系统响应速度：用户操作系统平均响应时间小于3秒。支持最大并发数为50。在维护方面整个应用软件系统应能够连续7\*24小时不间断工作，应用软件中的任一模块更新、加载时，在不更新与上下模块的接口的前提下，以不影响业务运转和服务为原则；应用软件具有较高的自动化程度，如：自动异常调度、自动故障告警、自动任务恢复等；应用软件具备相应容错手段，能容许操作人员的某些失误操作。在安全性上包括用户和权限管理，数据范围权限的控制，对支付通讯进行加密，防止数据篡改、攻击。

**2 术语表**

|  |  |
| --- | --- |
| 术语 | 术语解释 |
| HTML | 超文本标记语音，标准通用标记语言下的一个应用。“超文本”就是指页面内可以包含图片、链接，甚至音乐、程序等非文字元素 |
| CSS | 层叠样式表是一种用来表现HTML（标准通用标记语言的一个应用）或XML（标准通用标记语言的一个子集）等文件样式的计算机语言。 |
| javascript | JavaScript一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为JavaScript引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在HTML（标准通用标记语言下的一个应用）网页上使用，用来给HTML网页增加动态功能 |
| Tomcat | Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，属于轻量级应用[服务器](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8" \t "https://baike.baidu.com/item/tomcat/_blank)，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试JSP 程序的首选 |

## 3.设计概述

3.1 系统的复用计划：库、框架、模式、构件等方面的复用

本系统采用SpringBoot框架，创建独立的Spring应用程序。直接嵌入了Tomcat,自动配置Spring和第三方库，不需要XML配置。

MVC模式，Model View Controller，是模式-视图-控制器，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。Model（模型）表示应用程序核心（比如数据库记录列表）。View（视图）显示数据（数据库记录）。Controller（控制器）处理输入（写入数据库记录）。

MVC 模式同时提供了对 HTML、CSS 和 JavaScript 的完全控制。Model（模型）是应用程序中用于处理应用程序数据逻辑的部分。通常模型对象负责在数据库中存取数据。View（视图）是应用程序中处理数据显示的部分。通常视图是依据模型数据创建的。Controller（控制器）是应用程序中处理用户交互的部分。通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。

说明：在分析、设计、编码、评审等活动中未嵌入与复用有关的活动；软件过程中也未定义复用设计师、复用工程师的角色，故

创建可复用构件：无

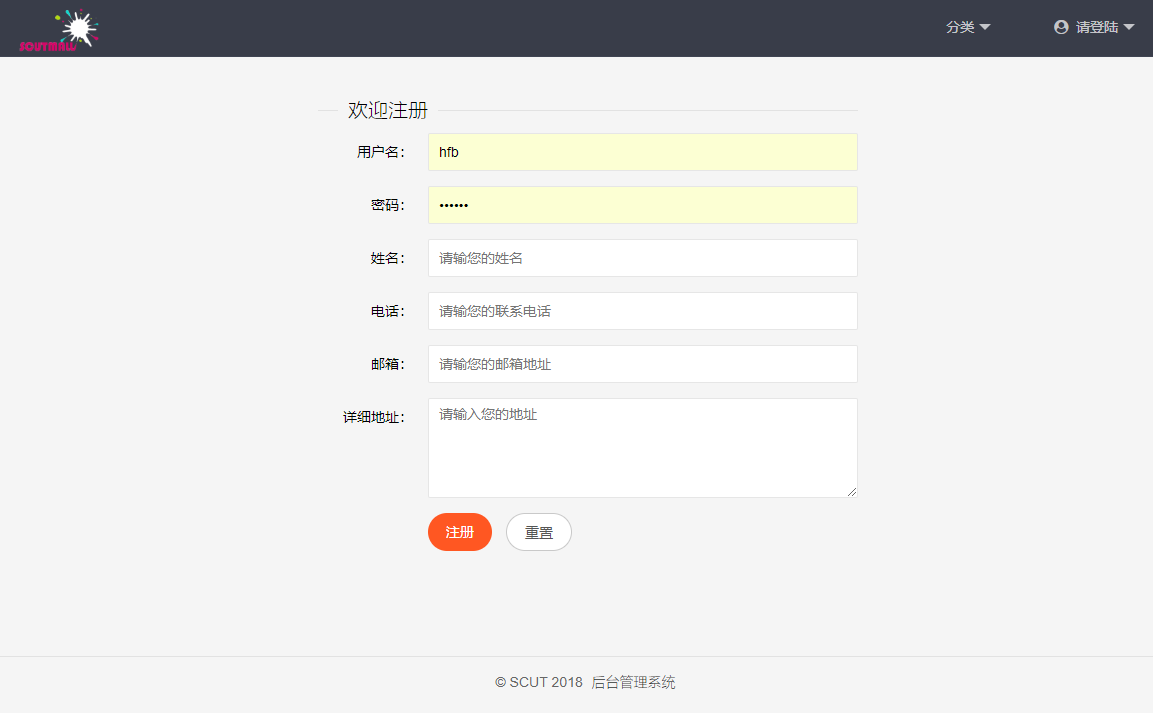
使用可复用构件：无

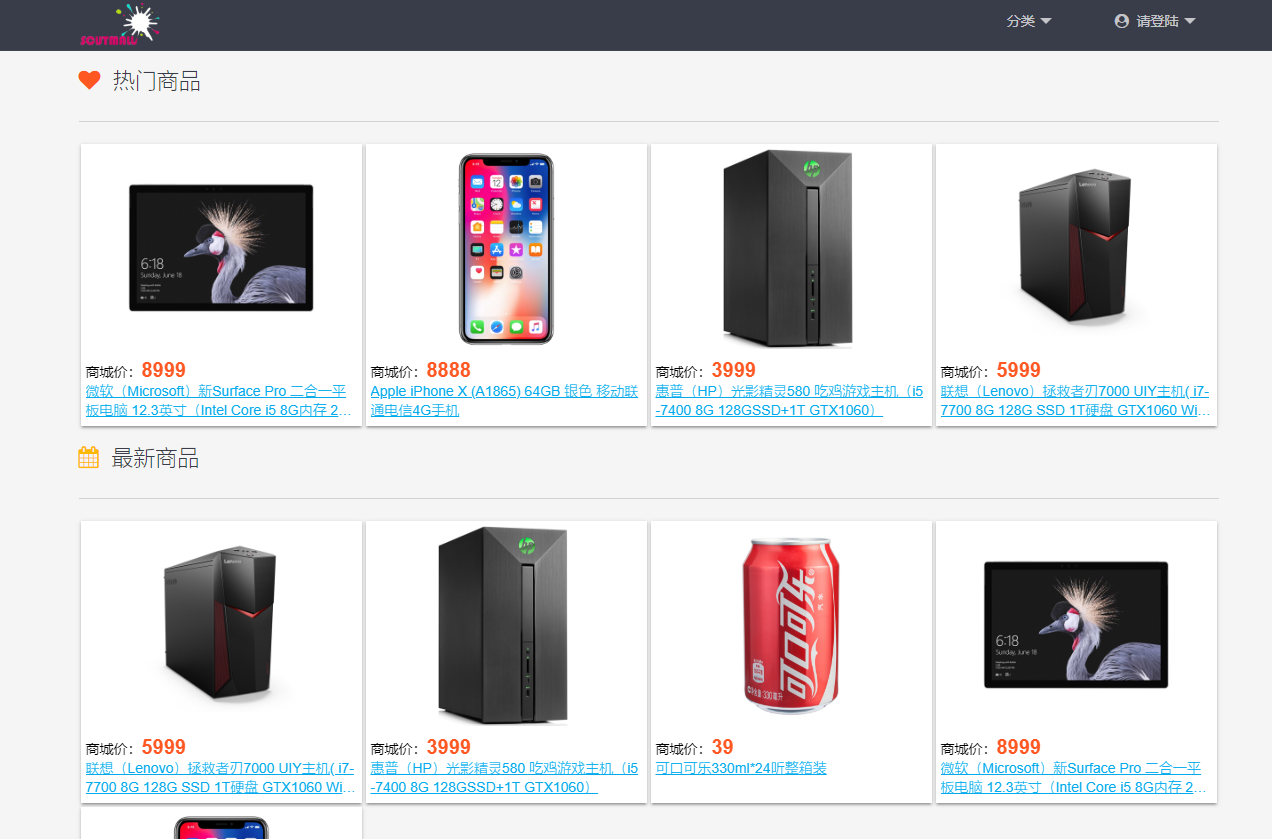
说明：在分析、设计、编码、评审等活动中未嵌入与复用有关的活动；软件过程中也未定义复用设计师、复用工程师的角色；项目规模较小，无复用需求；功能模块较为独立，无需复用构件支持。

3.2系统接口设计(根据概要设计进一步细化设计)

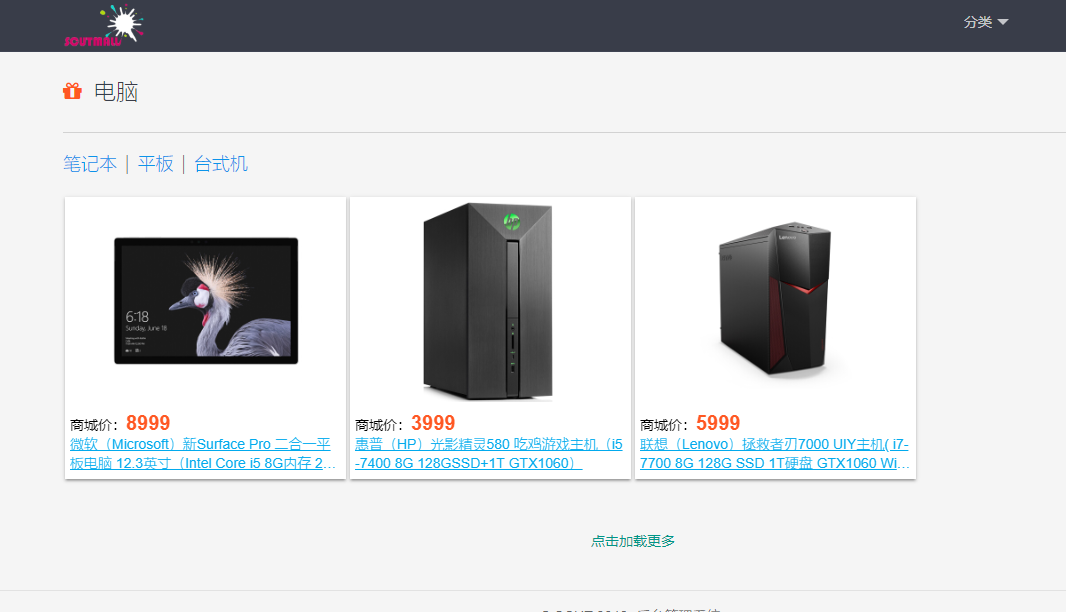
• 各种提供给用户的界面

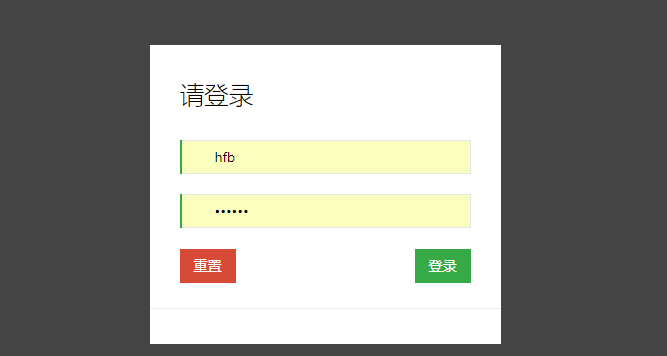




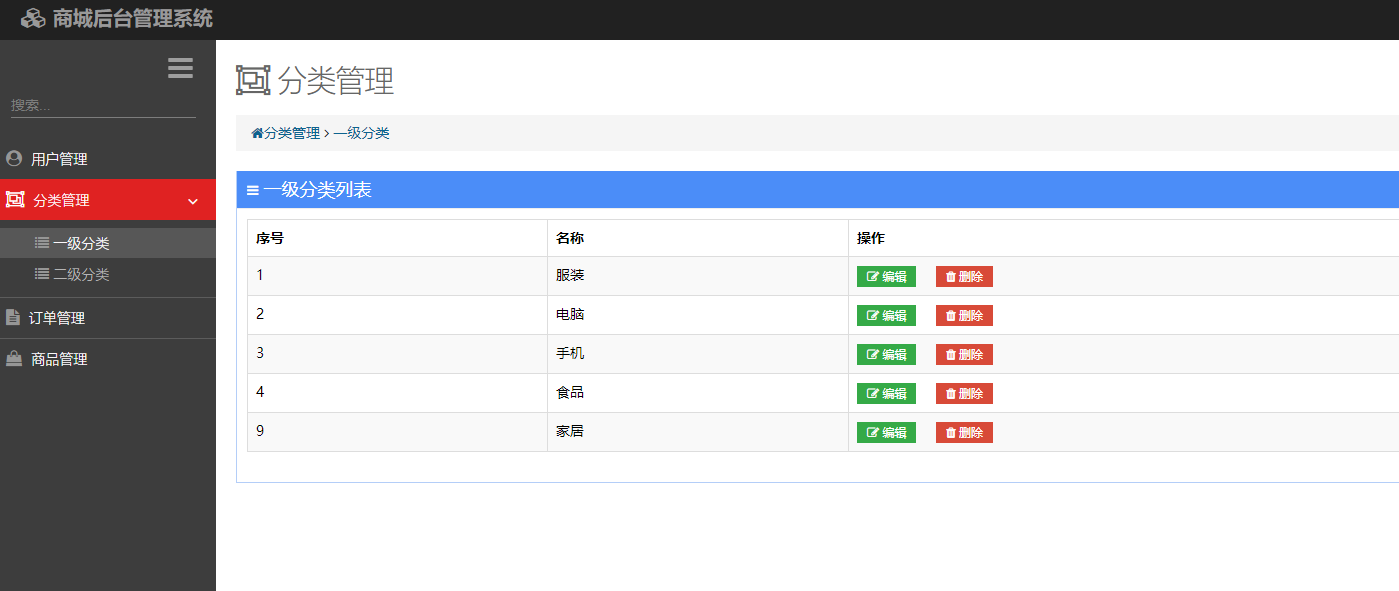




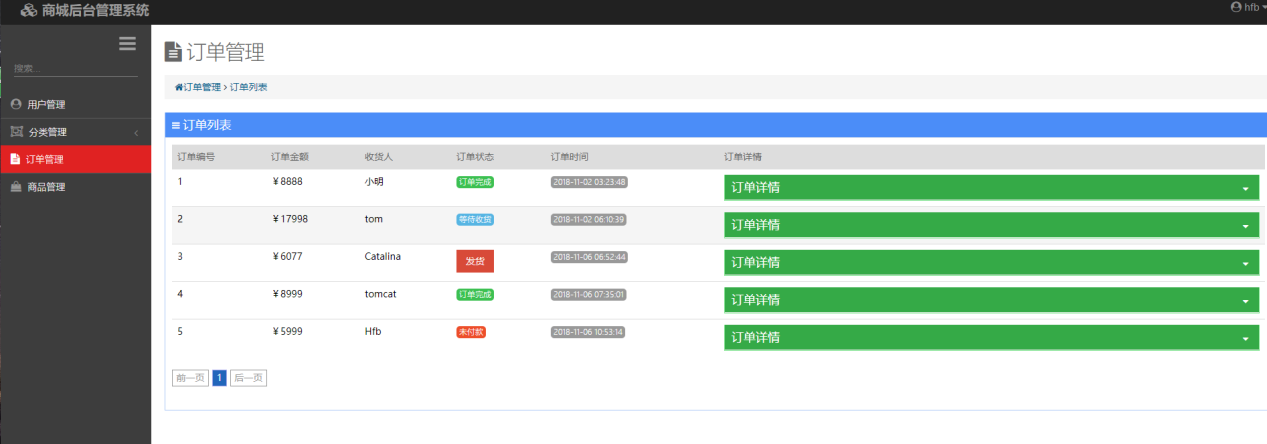


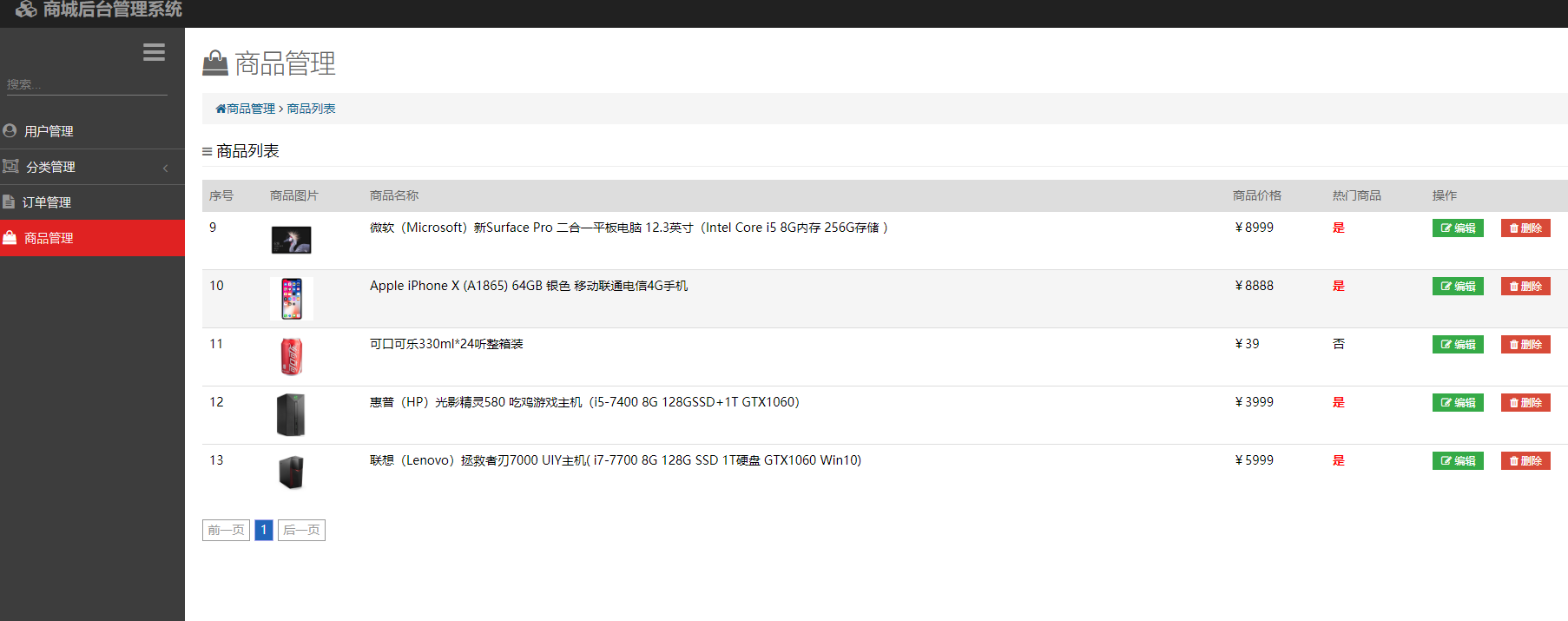












• 系统外部接口设计：与外部系统的交互设计

无

• 系统内部接口设计：各子系统、各模块间的接口设计

网站：

API：[http://localhost:8080/SCUTMall/index.html](http://localhost:8080/mall/index.html) 和http://localhost:8080/SCUTMall

页面说明：网站首页

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| product | Product | 显示商品的信息 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/user/toLogin.html

页面说明：网站登陆

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| username | String | 是 | 用户名 |
| password | String | 是 | 密码 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| user | User | 用户 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/user/toRegister.html

页面说明：网站注册

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| username | String | 是 | 用户名 |
| password | String | 是 | 密码 |
| name | String | 是 | 姓名 |
| phone | String | 是 | 电话 |
| email | String | 是 | 邮箱 |
| addr | String | 是 | 地址 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| user | User | 用户 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/category/category.html

页面说明：网站首页展示种类

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| cid | int | 是 | 选择分类 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| category | Category | 类别 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/product/get.html

页面说明：网站首页展示产品

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| Id | int | 是 | 产品id |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| product | Product | 产品 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/order/toList.html

页面说明：网站首页展示订单

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| order | Order | 订单 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/product/toCart.html

页面说明：网站首页加入购物车

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| name | String | 是 | 收货人 |
| phone | String | 是 | 电话 |
| addr | String | 是 | 地址 |
| request | HttpServletRequest | 是 | 请求 |
| response | HttpServletResponse | 是 | 响应 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| order | Order | 订单 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/product/listCart.do

页面说明：网站首页展示购物车

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| request | HttpServletRequest | 是 | 请求 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| orderItem | OrderItem | 订单明细项 |

API：[http://localhost:8080/](http://localhost:8080/mall/index.html)SCUTMall/product/delCart.do

页面说明：网站首页删除购物车

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| productId | Int | 是 | 产品id |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| msg | String | 删除提示信息（删除成功） |

网站后台管理：

API：<http://localhost:8080/SCUTMall/admin/toLogin.html>

页面说明：网站后台管理登陆

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| username | String | 是 | 用户名 |
| password | String | 是 | 密码 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| adminUser | AdminUser | 管理员 |

API：<http://localhost:8080/SCUTMall/admin/toIndex.html>

页面说明：网站后台管理首页

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| request | HttpServletRequest | 是 | 请求 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| User | User | 用户 |

API：<http://localhost:8080/SCUTMall/admin/user/toEdit.html>

页面说明：网站后台管理修改用户

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| username | String | 是 | 用户名 |
| password | String | 是 | 密码 |
| name | String | 是 | 真实姓名 |
| phone | String | 是 | 电话 |
| email | String | 是 | 邮箱 |
| addr | String | 是 | 地址 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| User | User | 用户 |

API：<http://localhost:8080/SCUTMall/admin/categories/toEdit.html>

页面说明：网站后台管理修改分类

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| Id | int | 是 | 一级分类ID |
| name | string | 是 | 一级分类名称 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| classification | Classification | 类目 |

API：<http://localhost:8080/SCUTMall/admin/categories/toList.html>

页面说明：网站后台管理展示分类

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| request | HttpServletRequest | 是 | 请求 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| classification | Classification | 类目 |

API：<http://localhost:8080/SCUTMall/admin/product/toEdit.html>

页面说明：网站后台管理修改产品

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| Id | int | 是 | 一级分类ID |
| name | string | 是 | 一级分类名称 |
| marketprice | double | 是 | 市场价格 |
| SCUTMallprice | double | 是 | 商场价格 |
| classification | Classification | 是 | 所属二级分类 |
| image | String | 是 | 商品图片 |
| desc | String | 是 | 商品描述 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| product | Product | 产品 |

API：<http://localhost:8080/SCUTMall/admin/product/toList.html>

页面说明：网站后台管理展示产品

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| request | HttpServletRequest | 是 | 请求 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| product | Product | 产品 |

API：<http://localhost:8080/SCUTMall/admin/order/getAll.html>

页面说明：网站后台管理展示订单

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| request | HttpServletRequest | 是 | 请求 |

返回结果：

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| order | Order | 订单 |

API：<http://localhost:8080/SCUTMall/admin/categories/toEdit.html>

页面说明：网站后台管理更改分类

请求参数：

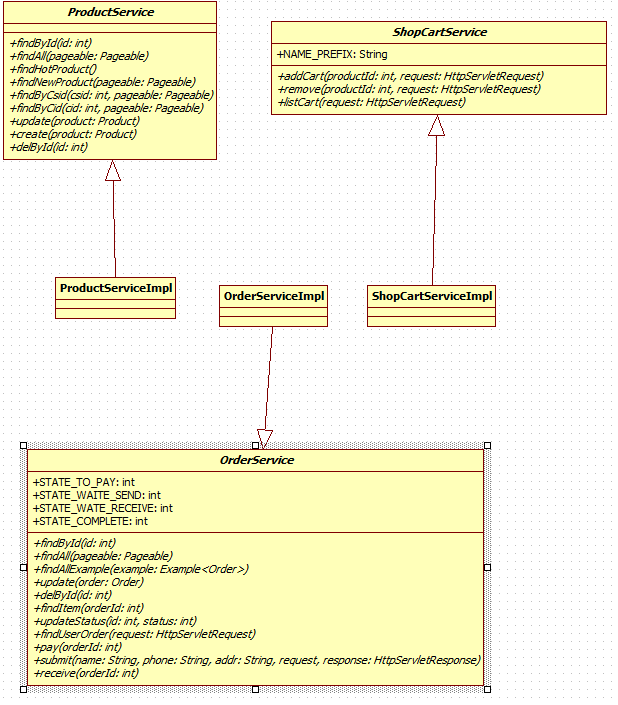
|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 是否必须 | 说明 |
| Id | int | 是 | 一级分类ID |
| name | string | 是 | 一级分类名称 |

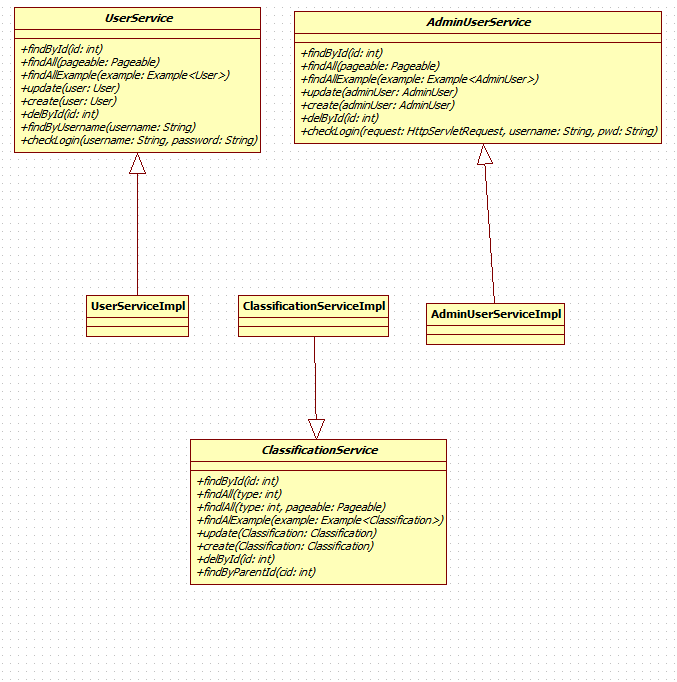
返回结果：

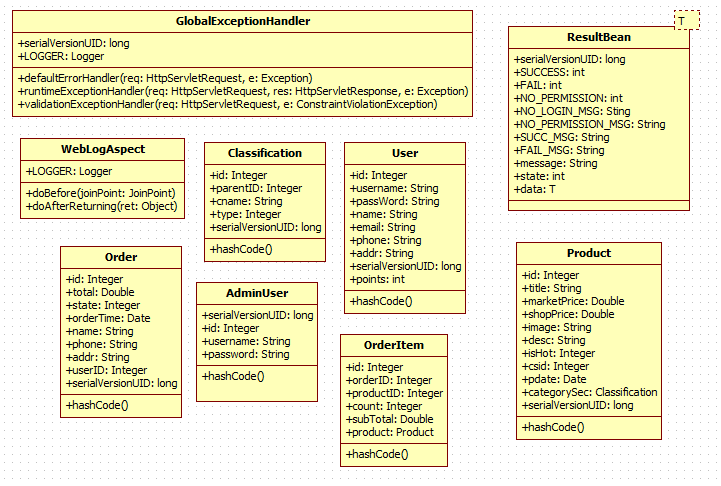
|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 说明 |
| classification | Classification | 类目 |

3.3 对象模型设计

3.3.1 对象模型:







3.3.2 对象描述(均省略get set方法)

GlobalExceptionHandler：用于统一的错误处理

属性：serialVersionUID：数据类型: long, 作用：序列化版本id

LOGGER: 数据类型：Logger，作用：用于记录错误时间处理日志

方法: ResultBean<String> defaultErrorHandler(HttpServletRequest req, Exception e)： 记录下错误信息和造成错误信息的访问源

void runtimeExceptionHandler(HttpServletRequest req, HttpServletResponse res, Exception e)：处理运行时错误

ResultBean<String> validationExceptionHandler(HttpServletRequest req, ConstraintViolationException e)：处理验证错误

WebLogAspect: 用于记录请求信息

属性：LOGGER：数据类型：Logger，作用：用于记录请求信息

startTime：数据类型：ThreadLocal<Long>，作用：用于记录请求开始时间

方法：void doBefore(JoinPoint joinPoint)：接受到请求处理前记录请求内容

void doAfterReturning(Object ret)：处理完请求后记录下返回的内容

ResultBean<T>：返回数据类型的接口

属性：serialVersionUID：数据类型：long，作用：序列化版本号

SUCCESS 数据类型：int，作用：定义成功代号为0

FAIL 数据类型：int，作用：定义失败代号为1

NO\_LOGIN\_MSG：数据类型：String，作用：定义未登录返回信息

NO\_PERMISSION\_MSG：数据类型：String，作用：定义无权限返回信息

SUCC\_MSG：数据类型：String，作用：定义成功返回信息

FAIL\_MSG：数据类型：String，作用：定义失败返回信息

message：数据类型：String，作用：返回信息

state：数据类型：int， 作用：当前状态

data：数据类型：T，作用：返回的数据

实体类共同具有方法与属性：

属性：serialVersionUID：数据类型：long，作用：序列化版本号

id：数据类型：Integer，作用：实体id号

方法：boolean equals(Object that)：用于对象之间进行比较，重载Object的equals函数

int hashCode()：获取该实体对象的hash值，先用质数31+id的hash值，在用31乘以这个结果，在+该实体类中另一个属性的hash值，依次处理完整个属性

String toString()：将该实体对象信息以json字符串形式给出

service接口(除了购物车服务外)共同具有方法：用于访问数据库

方法：findById(int id)：通过实体类对象id查询并返回要查询的实体类对象，

findAll(Pageable pageable)：分页查询所有的对应的实体类对象

List<T> findAllExample(Example<T> example)：按条件查询对应的实体类对象

update(T t)：更新该对象数据

create(T t)：将对象添加至数据库

delById(int id)：按照id号删除从数据库中删除该对象

AdminUserService：用于访问数据库管理者表的接口

特有方法：AdminUser checkLogin(HttpServletRequest request,String username, String pwd)：验证登陆并返回该管理者在数据库的信息

ClassificationService：用于访问数据库商品种类表的接口

特有方法：List<Classification> findByParentId(int cid)：利用该品类的上一级分类id查询商品并返回列表

UserService：用于访问数据库用户表的接口

特有方法：List<User> findByUsername(String username)：通过用户名查询并返回用户数据列表

OrderService：用于访问数据库用户表的接口

特有属性：STATE\_NO\_PAY：数据类型：int，作用：定义未支付状态代号为1

STATE\_WAITE\_SEND：数据类型：int，作用：定义等待发货状态代号为2

STATE\_WAITE\_RECEIVE：数据类型：int，作用：定义等待签收状态代号为3

STATE\_COMPLETE：数据类型：int，作用：定义订单完成状态为4

特有方法：List<OrderItem> findItems(int orderId)：通过订单项id查询到该笔订单中特定的订单项信息

pay(int orderId)：支付该订单，并更新订单信息状态

updateStatus(int id, int status)：更新订单状态信息

List<Order> findUserOrder(HttpServletRequest request)：查询用户的订单列表

submit(String name, String phone, String addr, HttpServletRequest request, HttpServletResponse response)：确认订单信息

receive(int orderId)：确认收货，更新订单状态信息

ProductService：用于访问数据库产品表的接口

特有方法：List<Product> findHotProduct()：查询热卖商品并返回列表

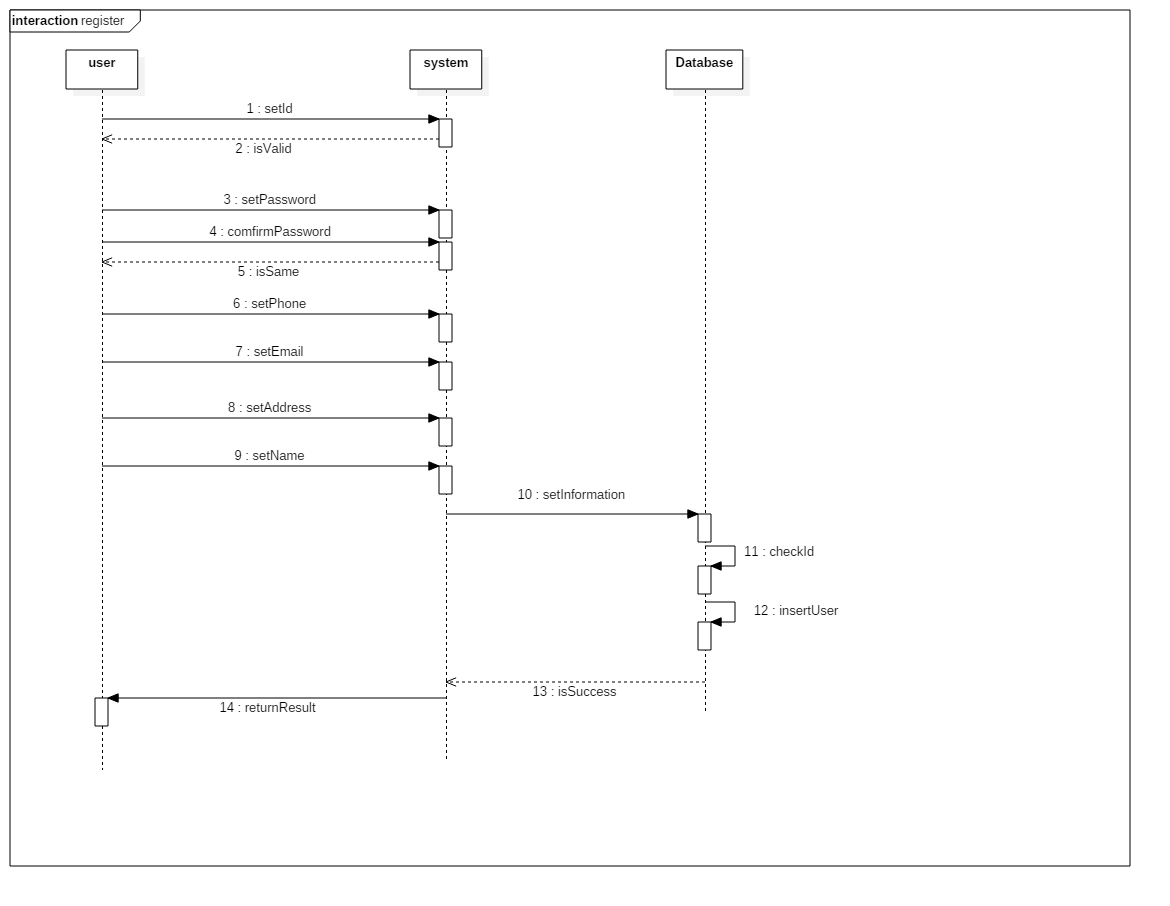
List<Product> findNewProduct(Pageable pageable)：查询最新商品并返回列表

List<Product> findByCid(int cid,Pageable pageable)：利用一级分类id查询商品并返回列表

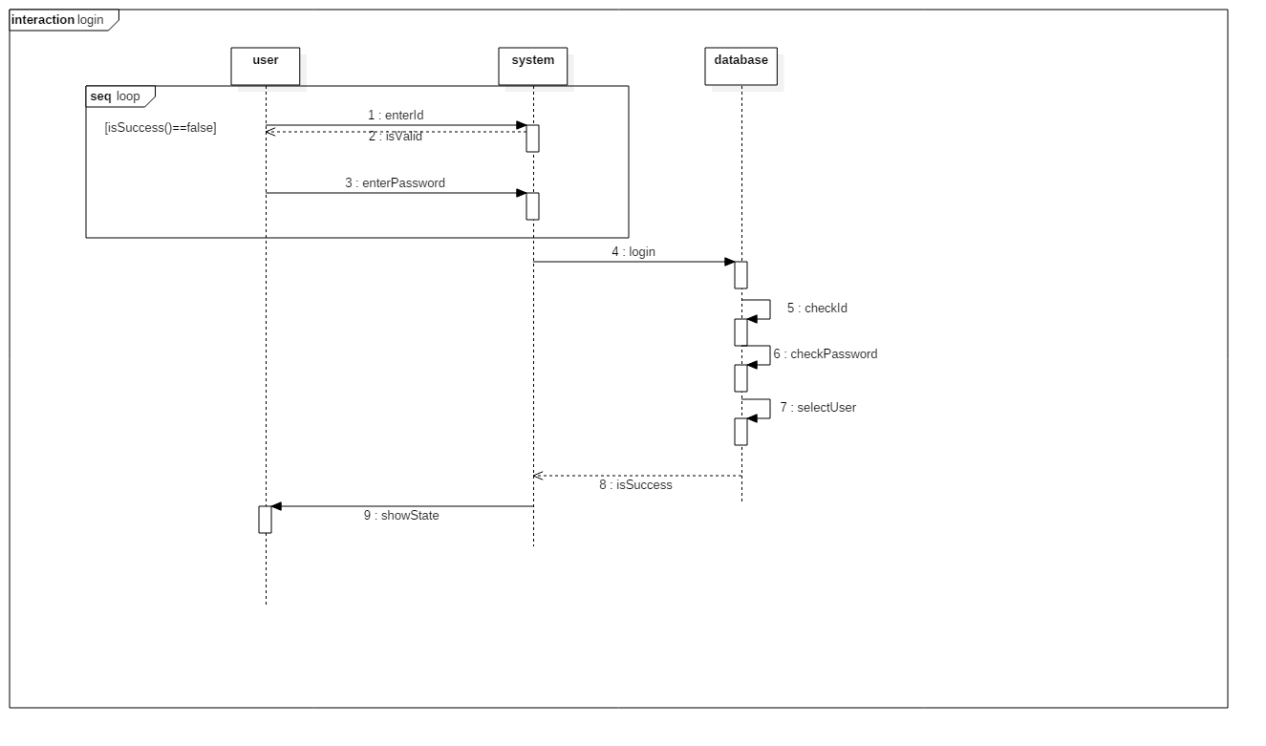
List<Product> findByCsid(int csid,Pageable pageable)：利用二级分类id查询商品并返回列表

## 3.4系统功能（用例）实现详细设计

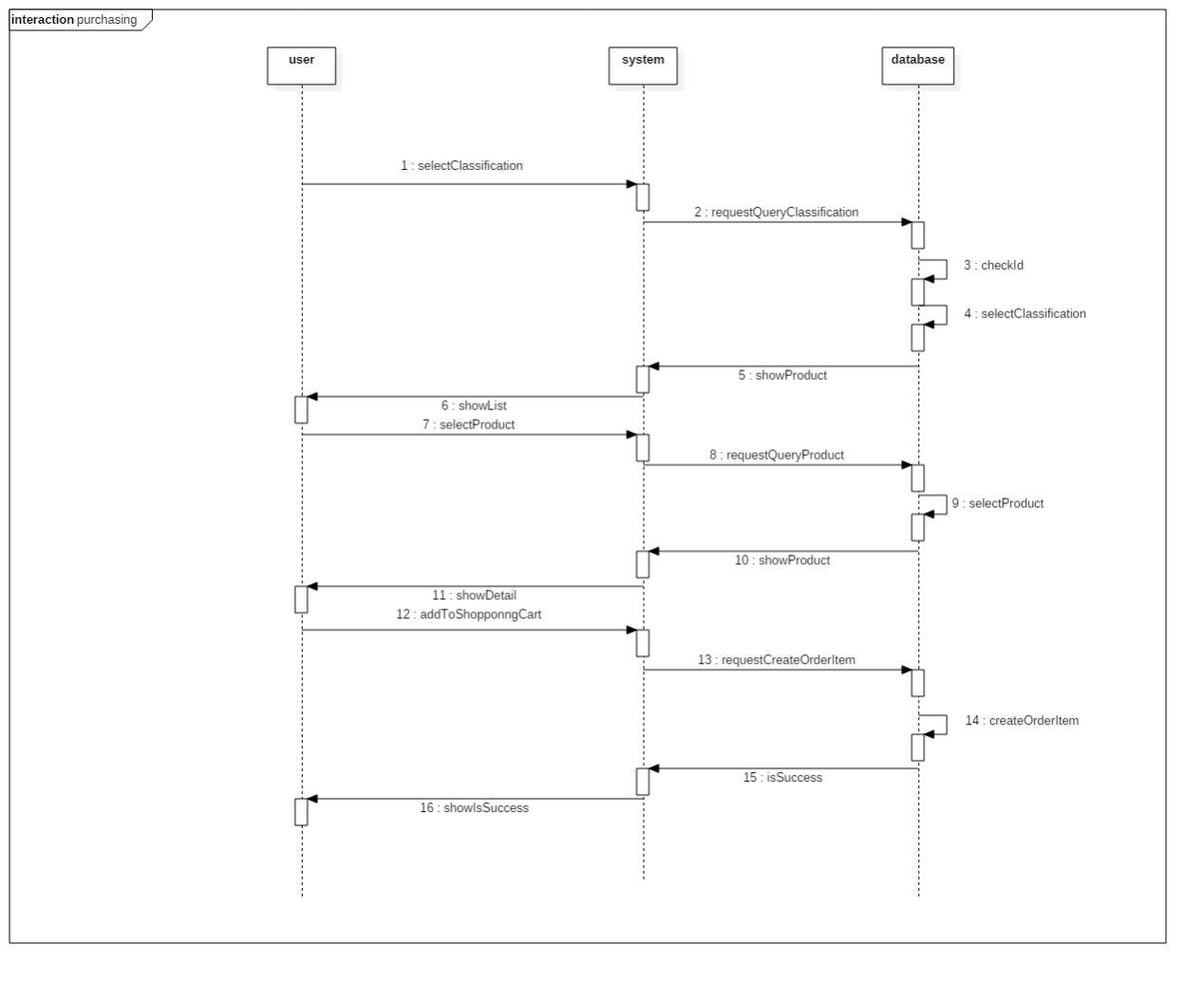
1. 顺序图：描述各种事件及事件发生的相对时间顺序
2. 用户注册



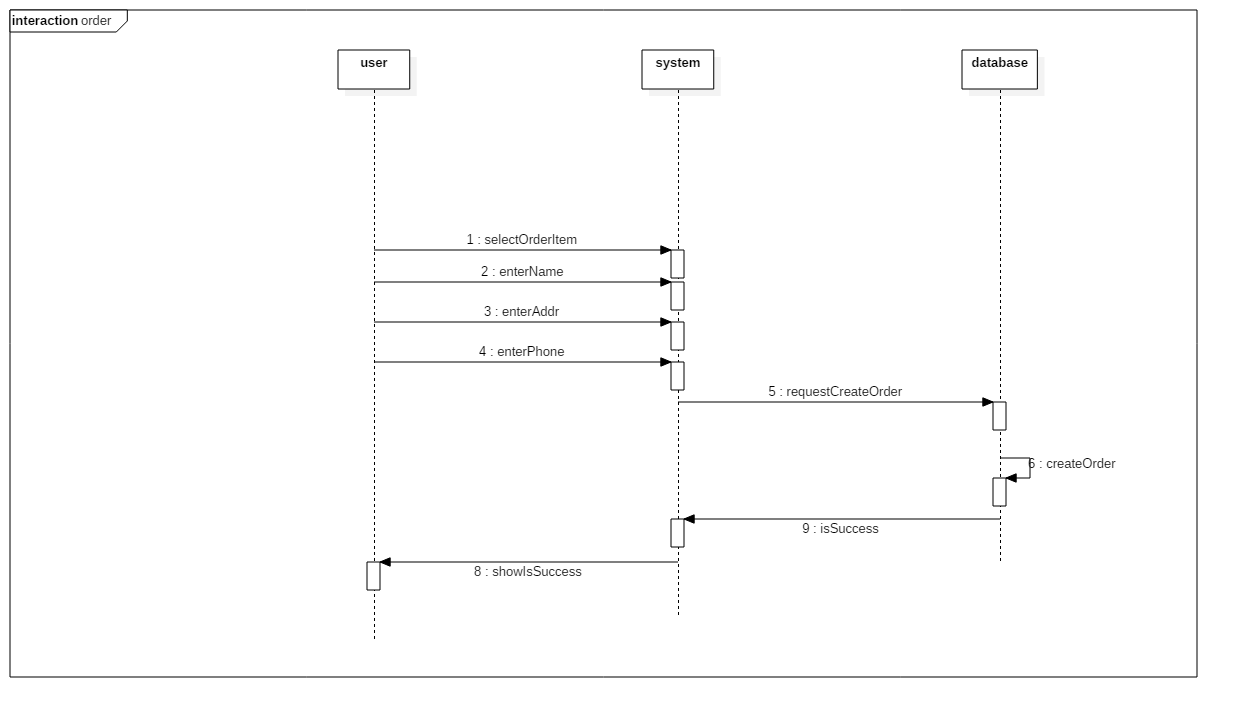
1. 用户登录



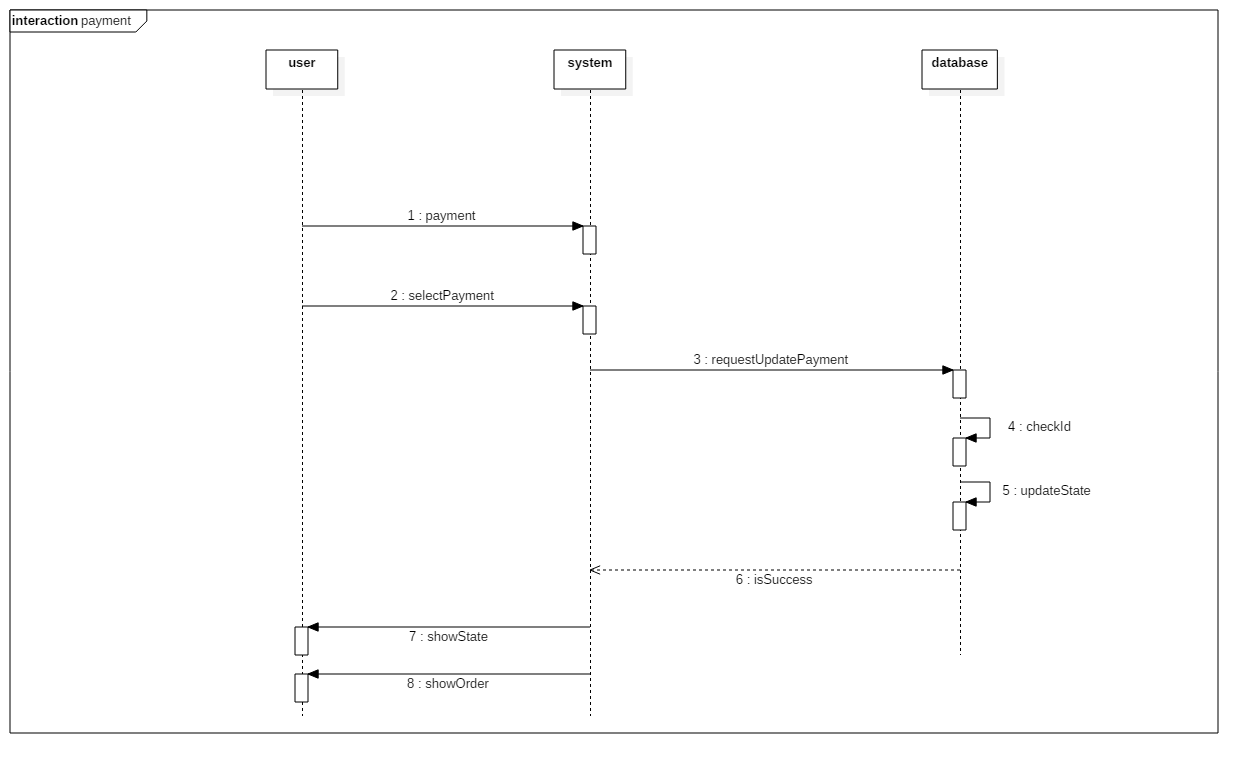
1. 用户浏览商品且加入购物车



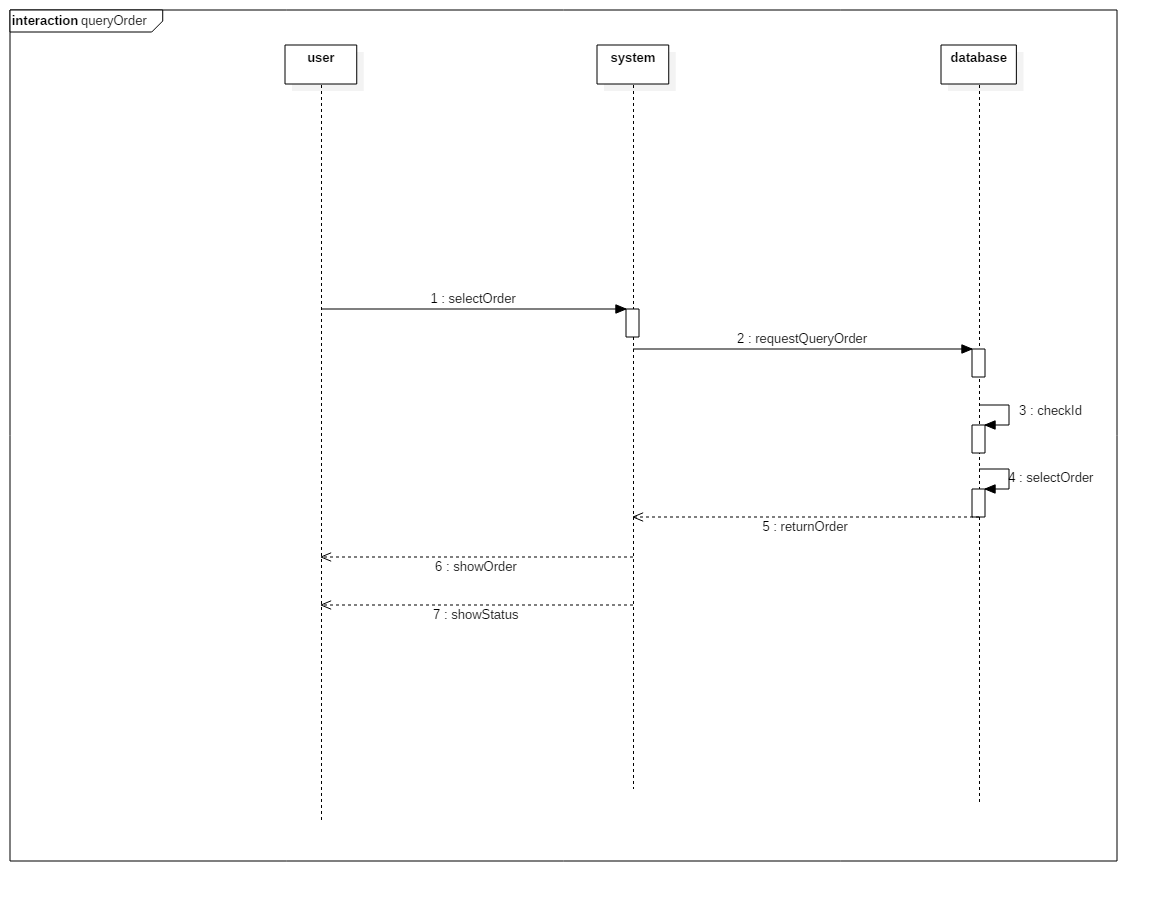
1. 用户下单



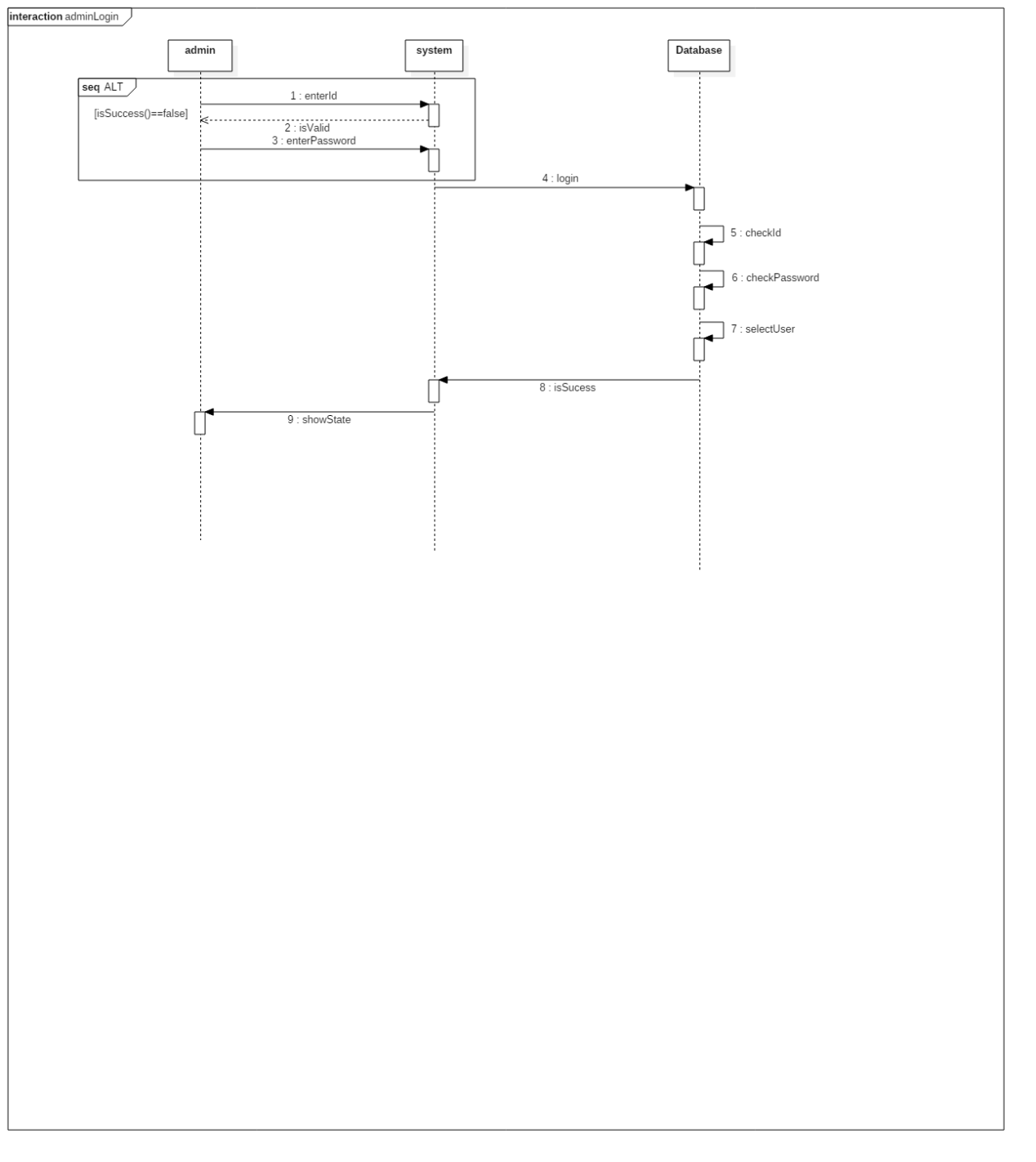
1. 用户付款



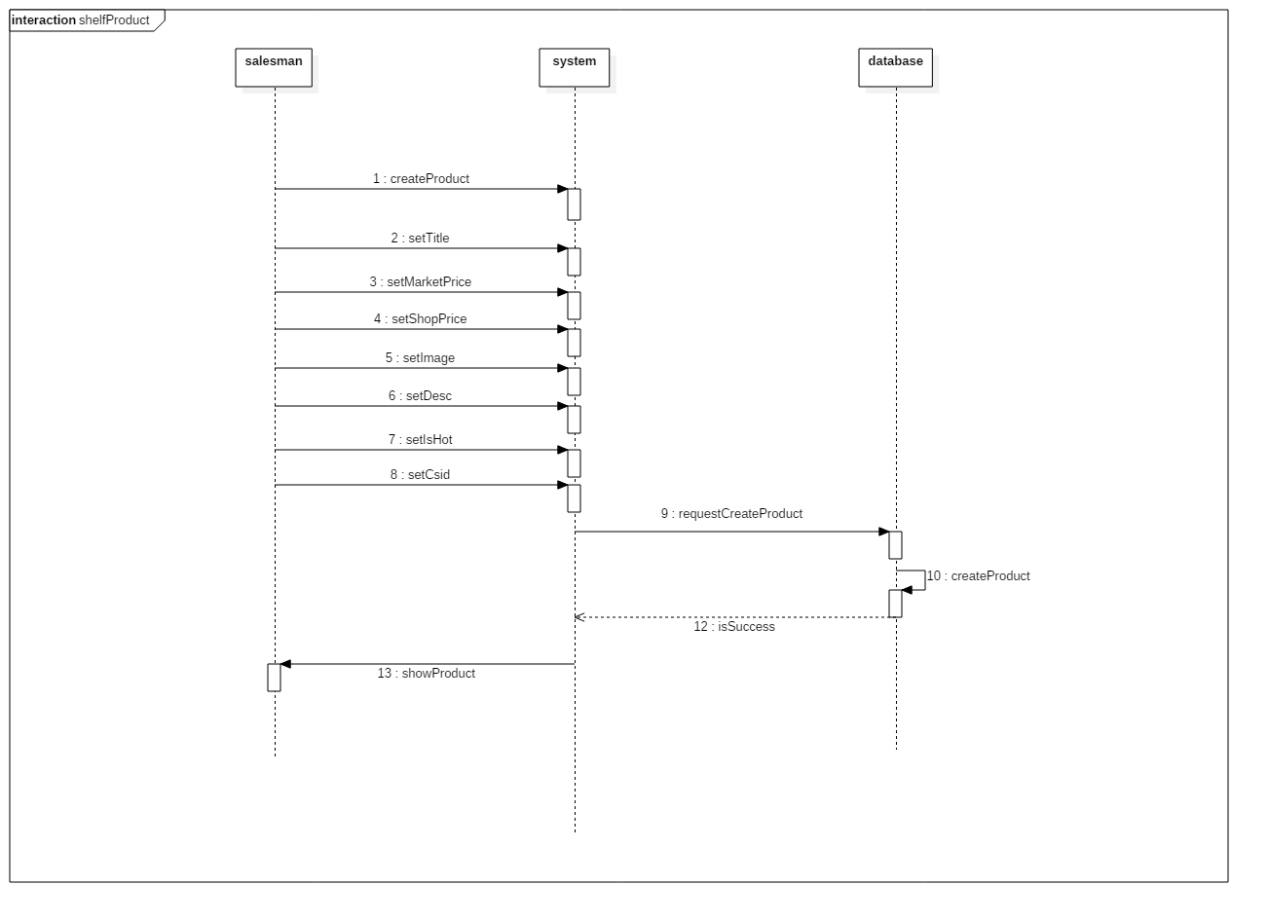
1. 用户查看订单



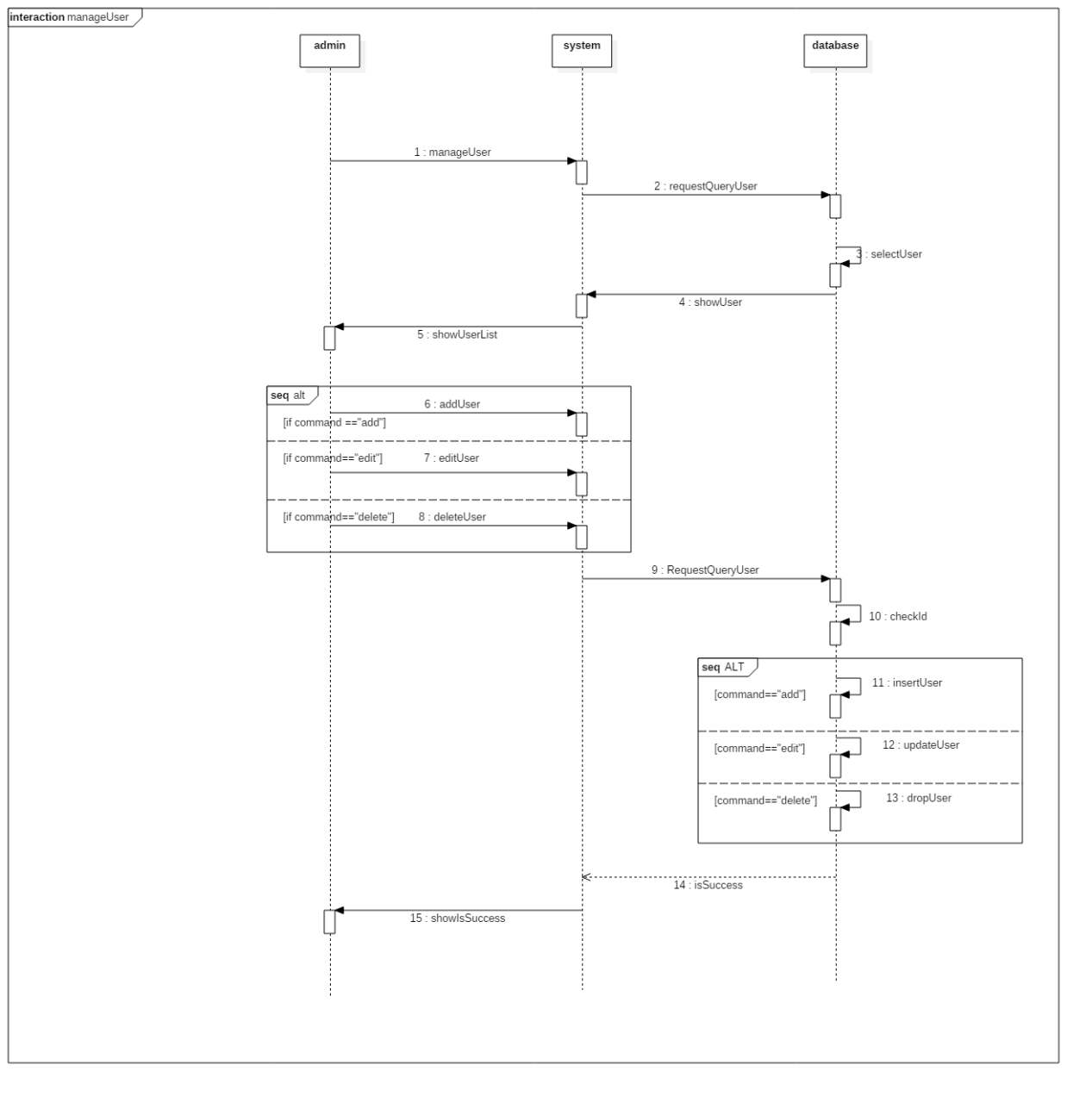
1. 管理员登录



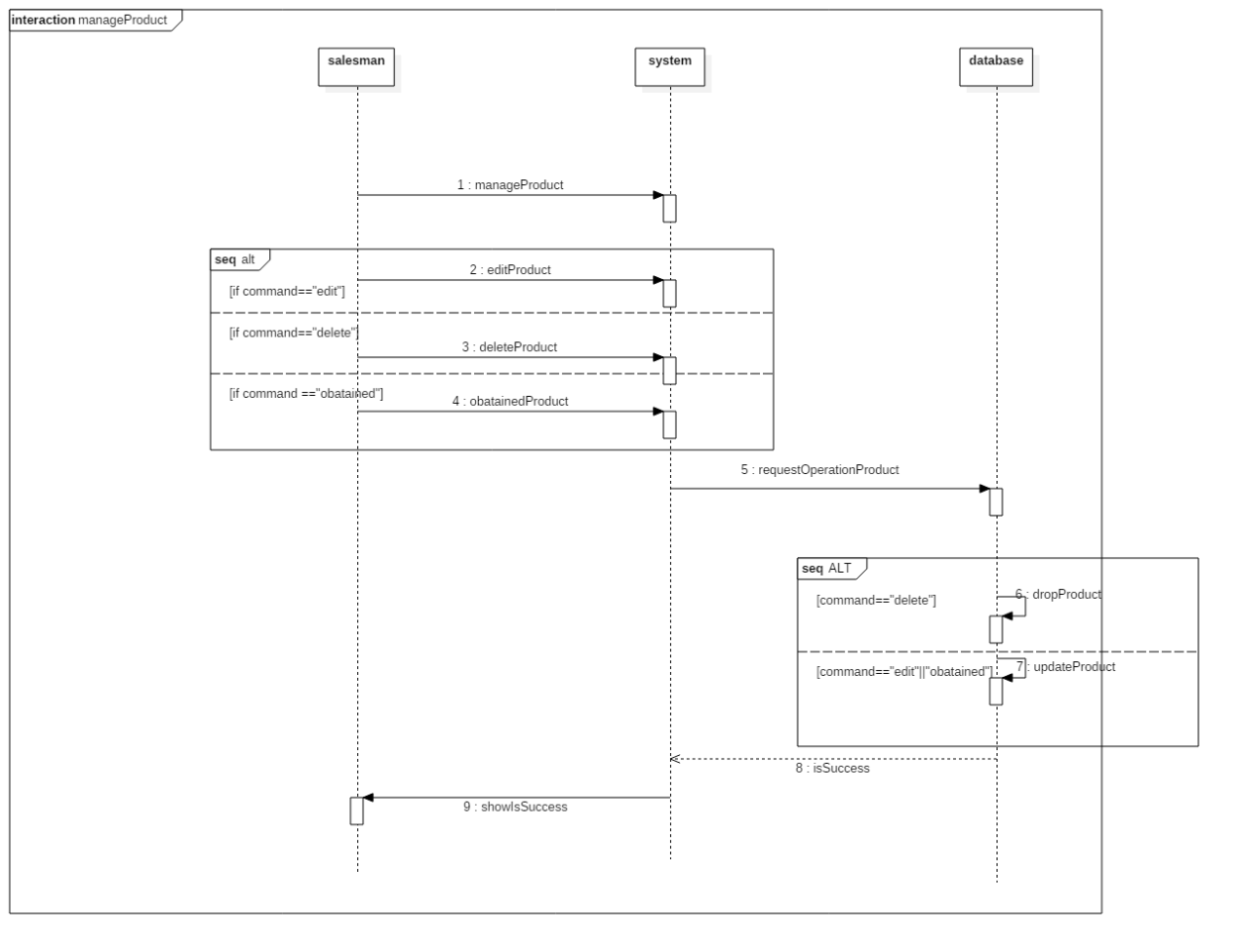
1. 业务员上架商品



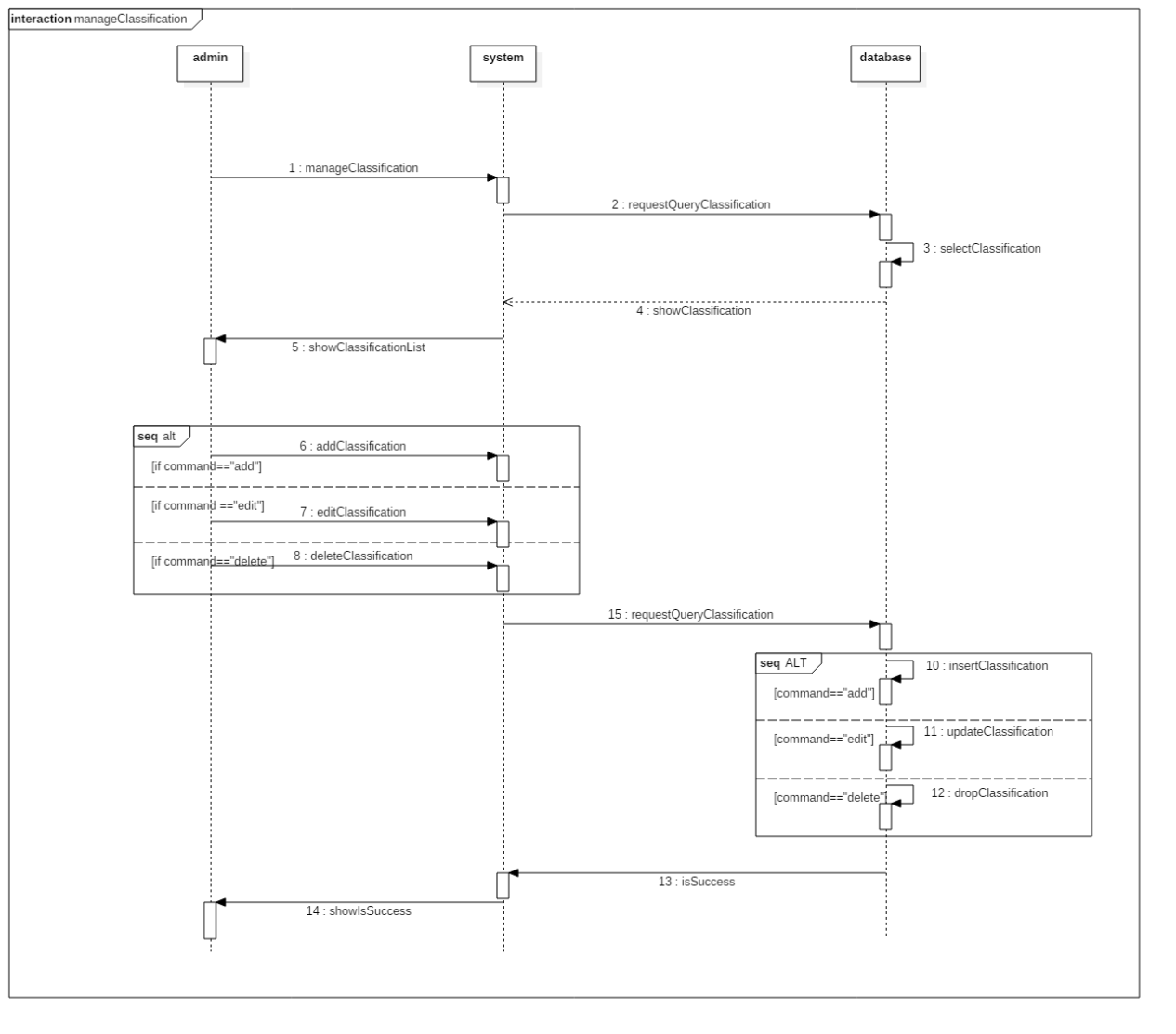
1. 管理员管理用户



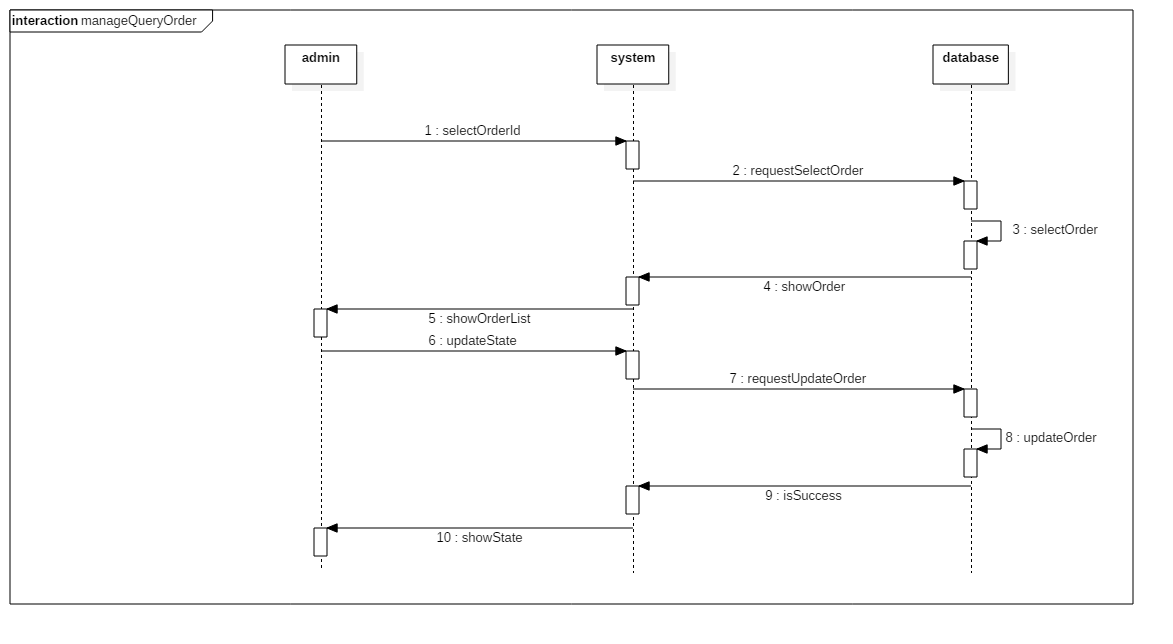
1. 管理员管理商品



1. 管理员管理分类

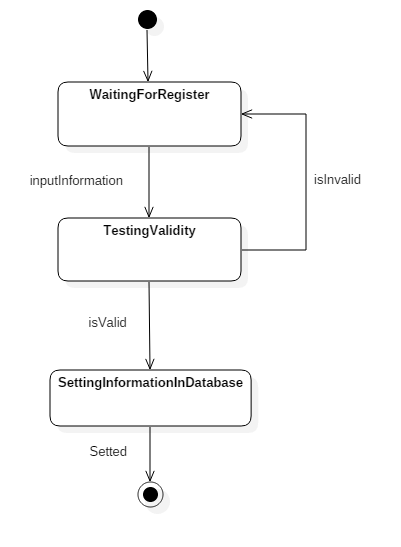


1. 管理员管理订单

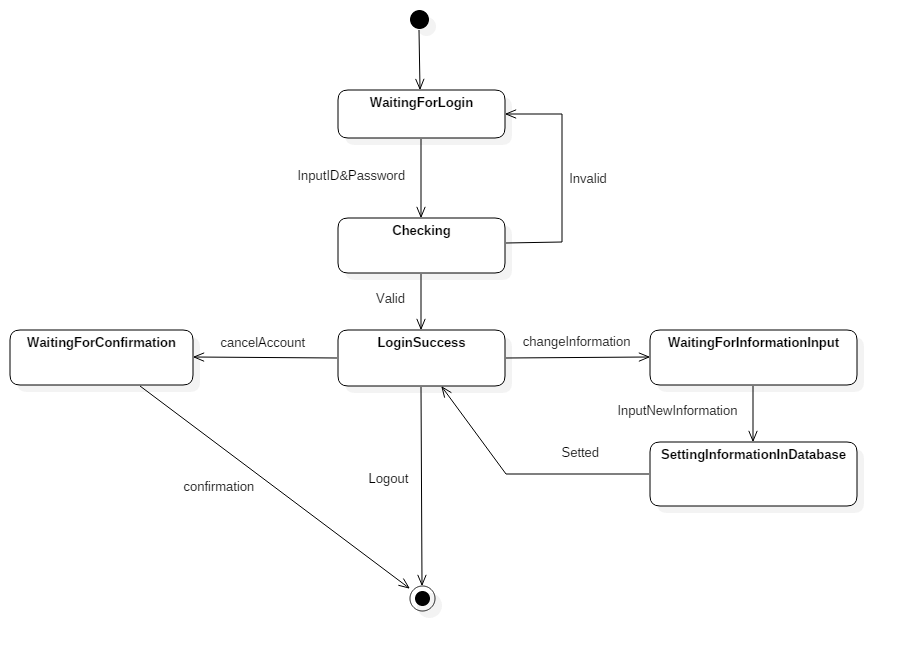


2.状态图（按需要画） 这部分的内容包括系统动态模型重要的部分的状图

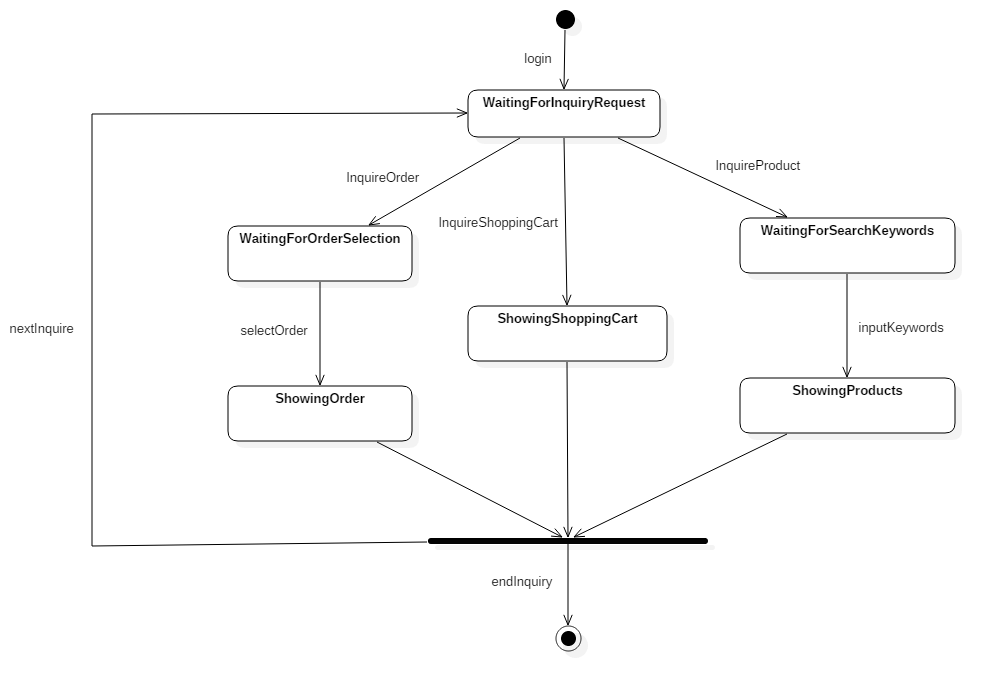
* 1. 用户\_注册



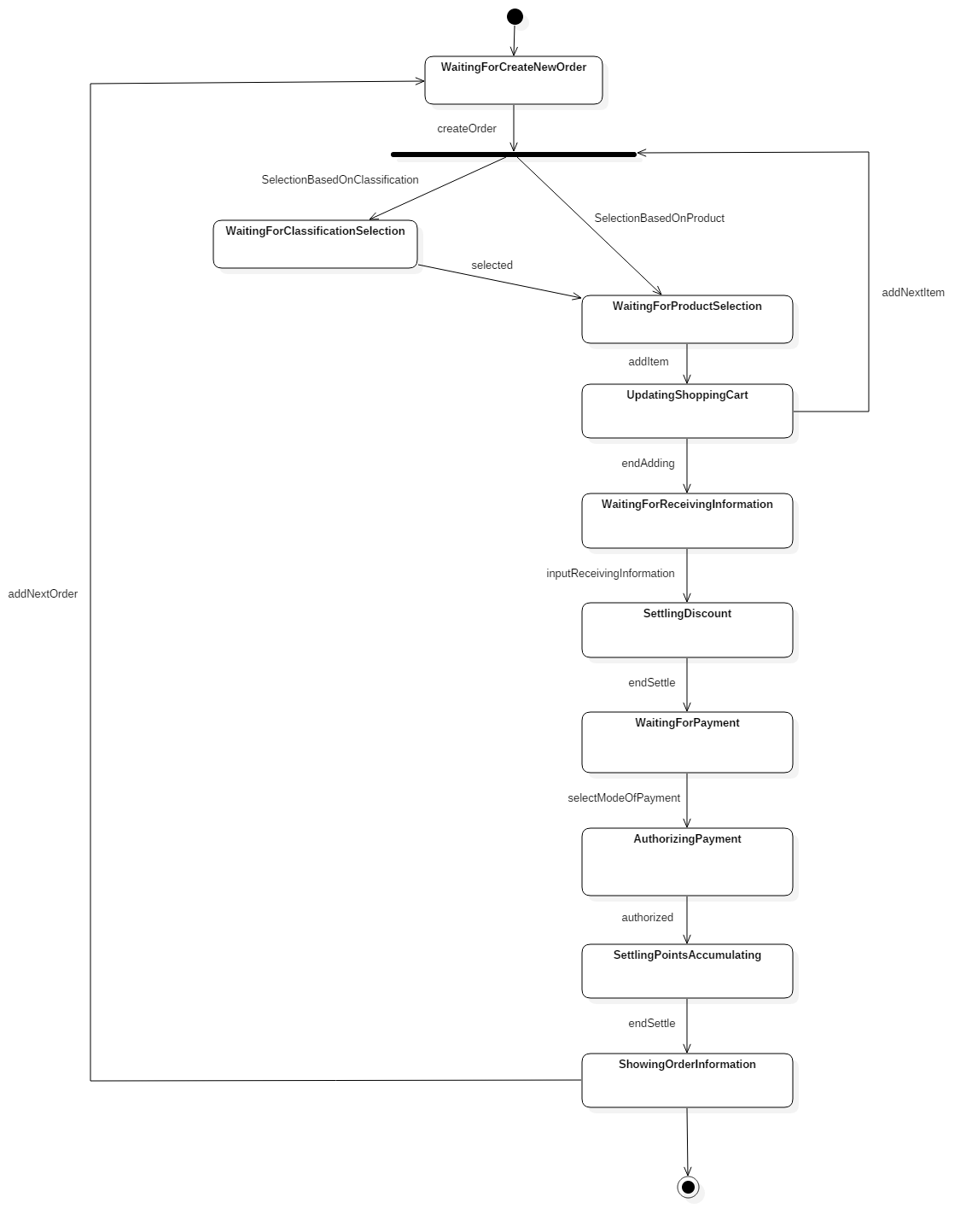
* 1. 用户\_账号操作



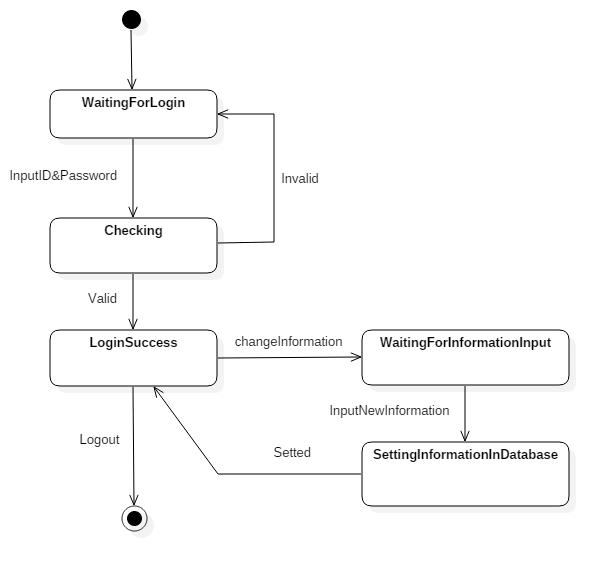
* 1. 用户\_查询操作



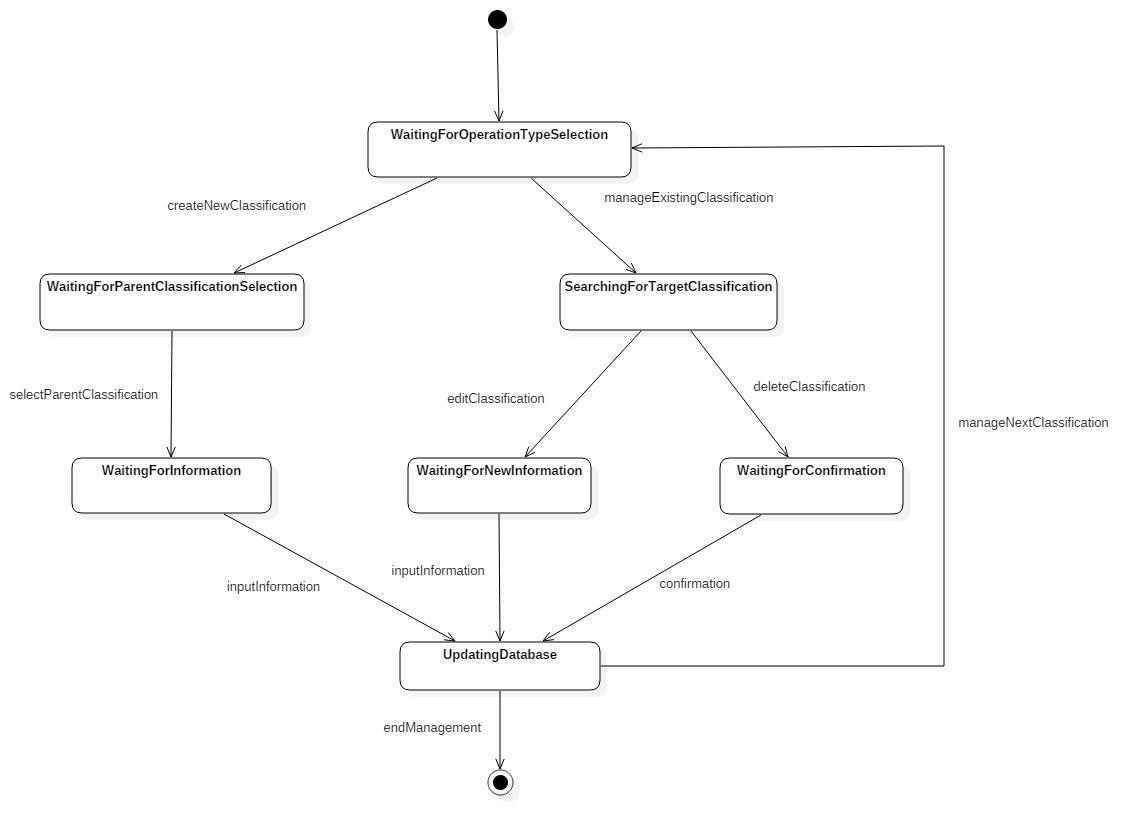
1.4 用户\_购物操作



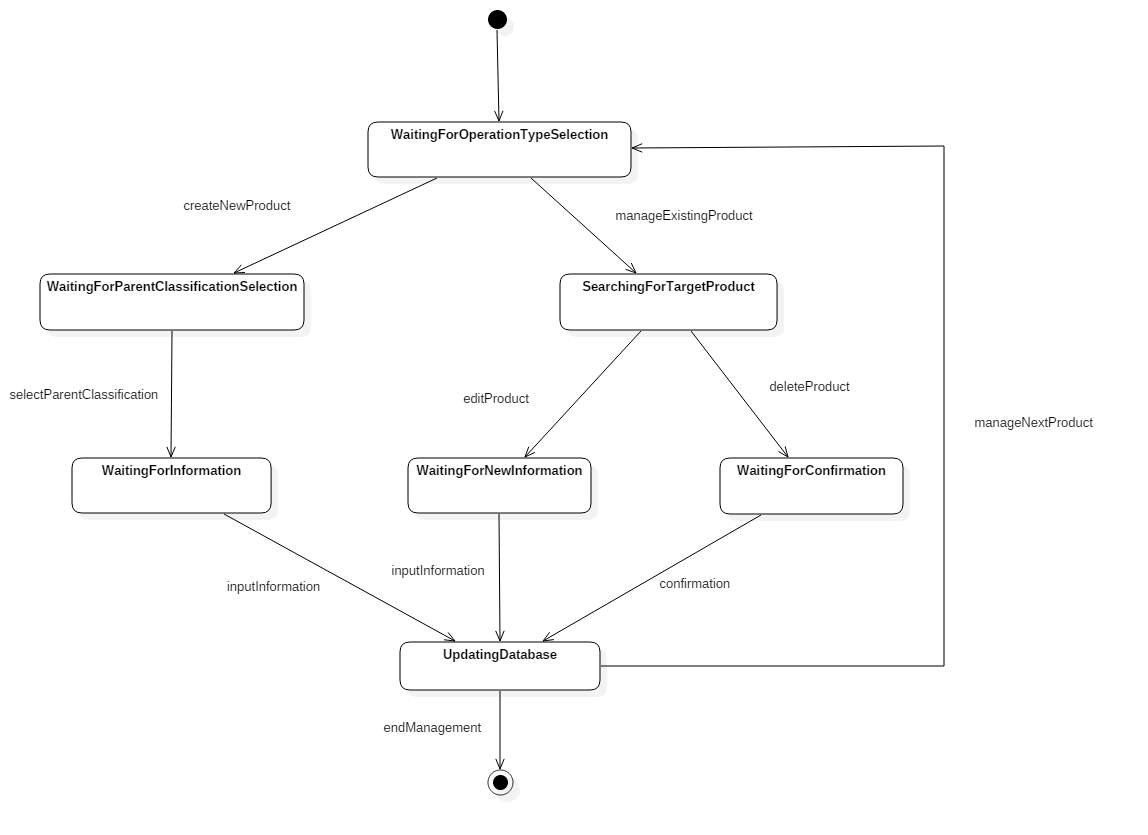
2.1 业务员\_账号操作



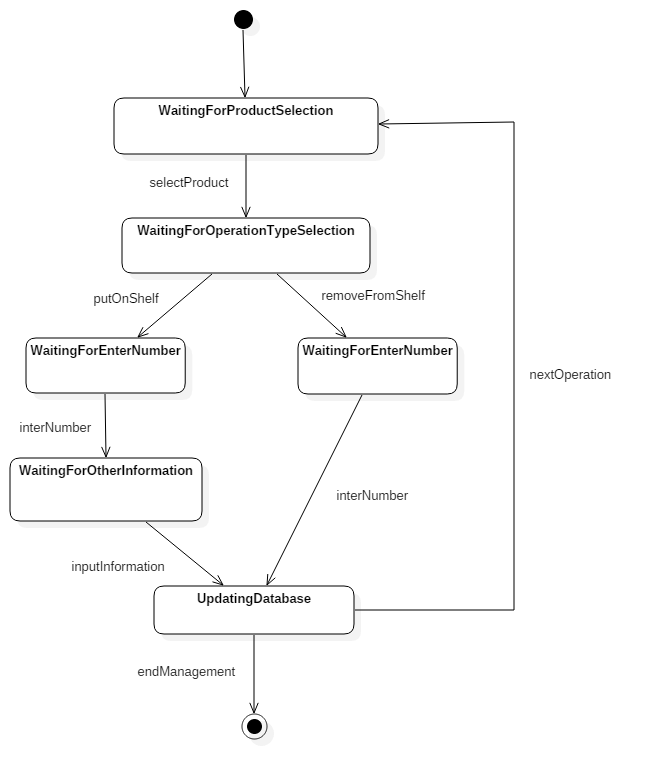
2.2业务员\_品类管理



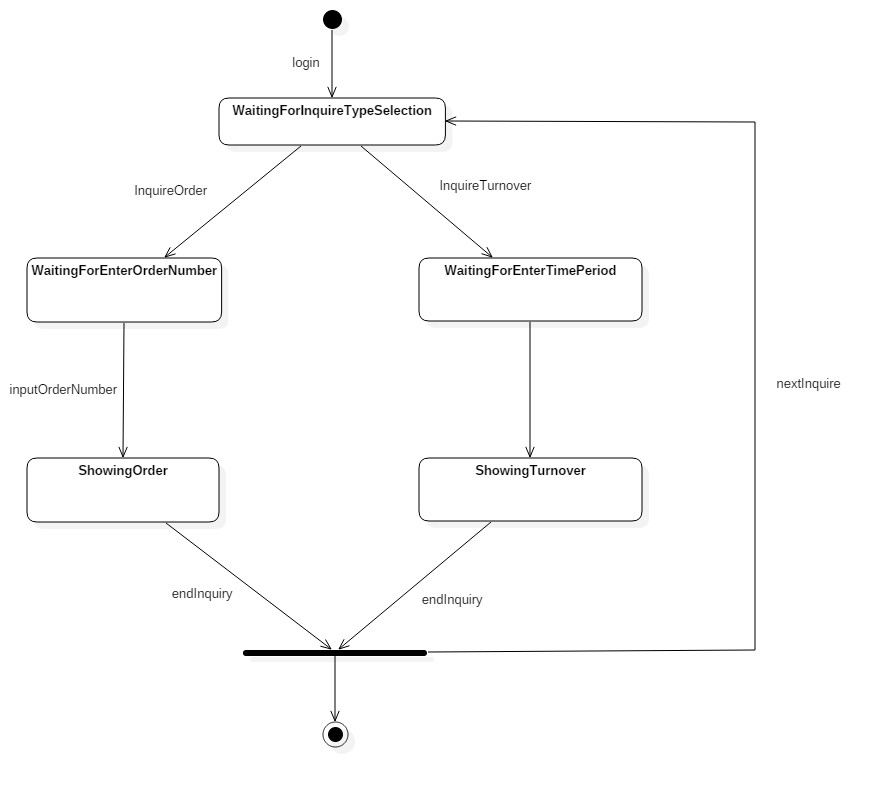
2.3业务员\_产品管理



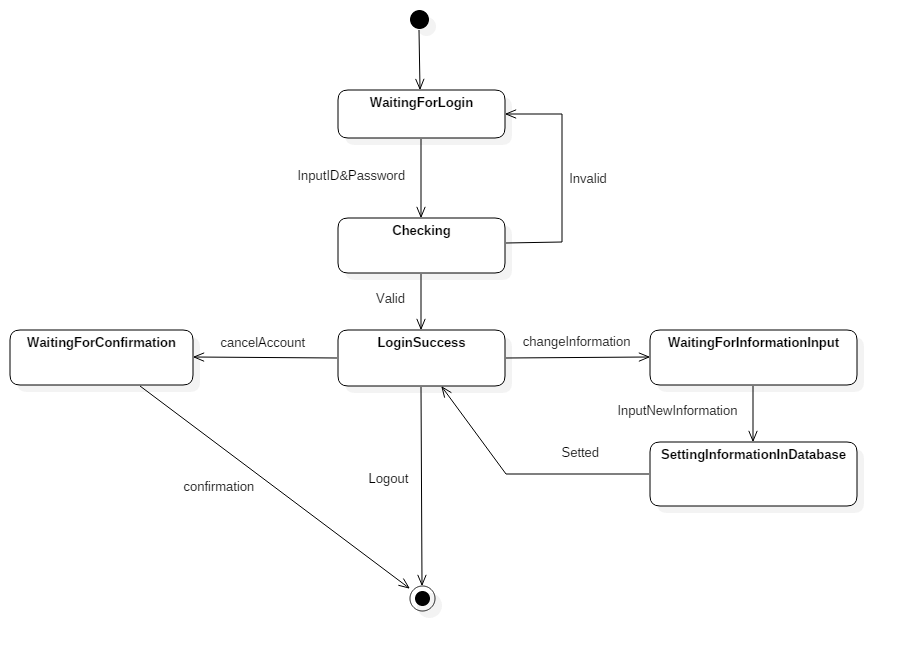
2.4 业务员\_货物上下架



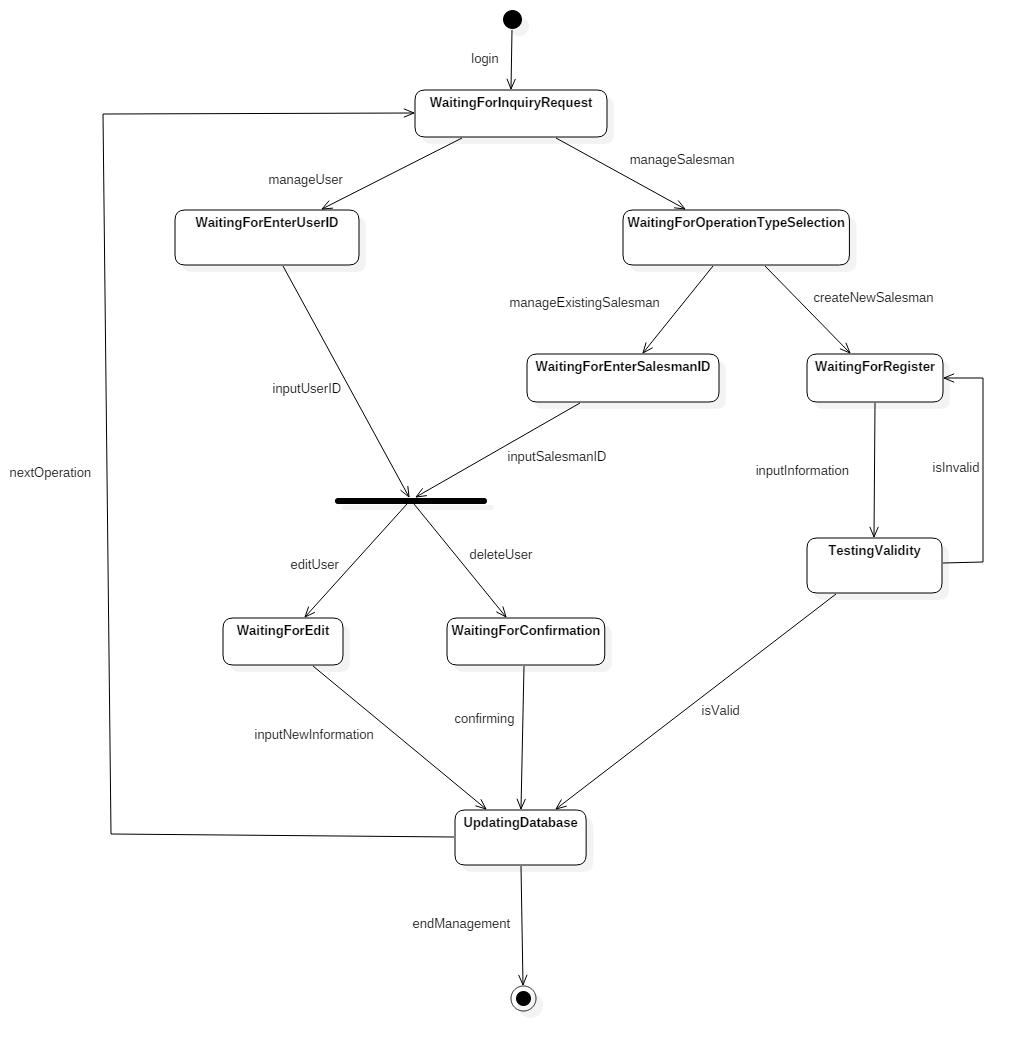
2.5 业务员\_查询操作



3.1 系统管理员\_账户操作



3.2 系统管理员\_用户与业务员管理



**3.5系统非功能设计**

**各个模块进行实现时需要同时考虑到相应非功能性设计的约束，以下为模块与非功能性设计的约束矩阵。**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **模块**  **约束** | **账户模块** | **商品管理模块** | **查询模块** | **购物模块** | **系统管理模块** | **统计模块** |
| **性能设计** | **√** | **√** | **√** | **√** |  |  |
| **维护性设计** | **√** | **√** | **√** | **√** | **√** | **√** |
| **易用性设计** | **√** | **√** | **√** | **√** | **√** |  |
| **兼容性设计** | **√** |  | **√** | **√** |  |  |
| **可拓展性设计** | **√** | **√** | **√** | **√** | **√** | **√** |
| **可靠性设计** | **√** | **√** | **√** | **√** | **√** |  |
| **灾备设计** |  | **√** |  |  | **√** |  |
| **安全性设计** | **√** |  |  | **√** | **√** |  |

**3.5.1 性能设计**

* + - 系统响应速度：用户操作系统平均响应时间小于3秒。
    - 支持最大并发数：100

**3.5.2 维护性设计**

* 整个应用软件系统应能够连续7\*24小时不间断工作，应用软件中的任一模块更新、加载时，在不更新与上下模块的接口的前提下，以不影响业务运转和服务为原则；
* 应用软件具有较高的自动化程度，如：自动异常调度、自动故障告警、自动任务恢复等；
* 应用软件具备相应容错手段，能容许操作人员的某些失误操作。

**3.5.3 易用性设计**

* 该系统提供基于图形化的友好管理界面；
* 符合IE界面风格：

 系统风格协调一致，体现人性化的友好性管理界面；

 符合用户习惯或者容易被用户接受的管理风格；

 具有容错能力，包括错误诊断和提示，并具有快速的系统反应的管理界面。

* 具有详细的、易懂的联机帮助，协助用户使用。

**3.5.4 兼容性设计**

* 对Chrome, Firefox, 360浏览器的进行兼容

**3.5.5 可拓展性设计**

* 对后期用户数量增大时的可拓展性的支持
* 对积分规则、活动结算的业务可拓展性的支持

**3.5.6 可靠性设计**

* 系统满足7\*24稳定运行需要。系统平均无故障时间（MTTF）大于4320小时（180天）；
* 系统容错性强，在外系统故障、用户非法操作、数据内容/格式出错的情况下，仍可正常运行平均失效间隔（MTBF）大于4320小时（180天）；
* 系统故障状态下恢复时间短，平均恢复时间（MTTR）小于10分钟；
* 对人工输入的数据以及来自不同接口的数据进行合法性检查，对错误数据进行自动纠错处理。

**3.5.7 灾备设计**

* 系统做到能支持定期的自动数据备份操作，还提供人工备份的操作。
* 提供多种数据输出格式，该输出格式可以方便快捷的倒回原来系统中。
* 真正做到使系统能在数据损坏，丢失等情况下将备份数据倒回，实现数据恢复。

**3.5.8 安全性设计**

* 用户和权限管理
* 数据范围权限的控制
* 对支付通讯进行加密
* 防止数据篡改、攻击