

# 软件工程课程项目实践报告

(SE224 微电子学院 2019 春)

## “Web 二手书交易平台” 小组

### 软件需求规格说明书

## 目 录

<b>1 导言 .....</b>	<b>6</b>
1.1 目的.....	6
1.2 范围.....	6
1.3 缩略语定义.....	6
1.4 参考文献.....	6
<b>2 总体描述.....</b>	<b>7</b>
2.1 产品愿景.....	7
2.2 产品特性.....	7
2.3 用户类型和特征.....	7
2.4 操作环境.....	8
2.5 设计和实施约束.....	8
<b>3 系统特性.....</b>	<b>9</b>
3.1 功能概览.....	9
3.2 功能详述.....	9
3.2.1 用户账号注册/登陆（全组） .....	9
3.2.2 商品增删/编辑（杨文曦） .....	12
3.2.3 商品检索/排序（张佳玲） .....	16
3.2.4 用户聊天（丁维德） .....	19
3.2.5 订单生成/处理（杜培栋） .....	21
3.2.6 订单评价（朱重熙） .....	25
3.2.7 用户信息管理（Karen） .....	28
<b>4 外部接口需求.....</b>	<b>32</b>
4.1 用户界面.....	32
4.2 硬件接口.....	32
4.3 软件接口.....	32
4.4 通信接口.....	32
<b>5 其他非功能需求.....</b>	<b>33</b>
5.1 性能需求.....	33
5.2 安全需求.....	33
5.3 软件质量属性.....	33

# 1 引言

## 1.1 目的

本文档是对本小组着手开发的“二手书交易 Web 平台”的一个概括描述，旨在定义本项目的系统目标、系统特征及设计约束等总体要求，使软件开发工作更为具体。作为软件开发人员对本项目的共同理解，可以帮助团队更好地开发、管理和维护本项目。

## 1.2 范围

本文档适用于本项目开发者和想要详细了解项目开发计划的人。除在系统目标里详细阐述了项目目标外，其他部分更加偏向于有关项目开发的介绍。

## 1.3 缩略语定义

表 1：缩略语定义

术语、缩略语	解释
JAccount	上海交通大学校内教师/学生/工作人员使用的校内统一网络账户。
React	一个为数据提供渲染为 HTML 视图的开源 JavaScript 库。

## 1.4 参考文献

- [1] Facebook Inc: React, A JavaScript library for building user interfaces, <https://reactjs.org>;
- [2] sofish: 中文网页重设与排版 - GitHub, <https://github.com/sofish/typo.css>.

## 2 总体描述

### 2.1 产品愿景

对于上海交通大学闵行校区校内学生，目前存在校内二手书线上/线下交易的不便问题。我们提供的“二手书交易 Web 平台”是一个 Web 平台，它能够实现校内二手书的便捷交易。不同于以往的二手书交易平台，比如“孔夫子旧书网”或校内二手书店，我们的产品可以实现近距离交易，可以免去运费和运输时间，也不需要再在实体店面拥挤地翻找仓库中想要的书，而且符合我们小组希望将项目 and 实际学习生活结合起来的愿景，更加具有价值。

### 2.2 产品特性

表 2：产品特性列表

<p><b>用例名：</b>用户账号注册/登陆；</p> <p><b>用例概述：</b>注册须填写用户名、密码，重复密码，手机号/邮箱，并实现格式检验；账号登陆需实现封禁功能，若被封禁，会显示相关提示信息。</p> <p><b>优先级：</b>基础功能之一，高优先级。</p>
<p><b>用例名：</b>商品增删/编辑；</p> <p><b>用例概述：</b>利用数据库实现管理（上架/下架/更改/删除）商品并实现各种关键词检索商品，选中某本书后，利用 Ajax 方式显示详细信息；信息包括书名/作者/封面照片/版权页照片/瑕疵处照片/ISBN 号/库存量/其他信息。</p> <p><b>优先级：</b>基础功能之一，中优先级。</p>
<p><b>用例名：</b>商品检索/排序；</p> <p><b>用例概述：</b>利用数据库实现商品关于特定关键词的检索和排序。选中某本书后，利用 Ajax 方式显示详细信息；信息包括书名/作者/封面照片/版权页照片/瑕疵处照片/ISBN 号/库存量/其他信息；</p> <p><b>优先级：</b>基础功能之一，中优先级。</p>
<p><b>用例名：</b>用户聊天；</p> <p><b>用例概述：</b>实现买家和卖家利用页面聊天功能相互沟通；</p> <p><b>优先级：</b>基础功能实现之后，低优先级。</p>
<p><b>用例名：</b>订单生成/处理。</p> <p><b>用例概述：</b>用户可以将书添加至购物车，点击购买后实现生成订单/确认。</p> <p><b>优先级：</b>基础功能之一，高优先级。</p>
<p><b>用例名：</b>订单评价；</p> <p><b>用例概述：</b>实现交易双方相互评价；</p> <p><b>优先级：</b>中优先级。</p>
<p><b>用例名：</b>用户信息管理。</p> <p><b>用例概述：</b>可以实现禁用/解禁用户，可以禁止用户登陆。</p> <p><b>优先级：</b>基础功能之一，高优先级。</p>

### 2.3 用户类型和特征

表 3：用户类型和特征

用户类型	使用本系统的目标
买家	从卖家处查到并购买所需的商品。
卖家（管理员）	将自己的商品卖给相应的买家，并对书籍进行管理。

## 2.4 操作环境

Web 页面，各操作系统各浏览器都可访问。

## 2.5 设计和实施约束

- 拟采用编程语言：Java / JavaScript;
- 前端管理：WebPack;
- 用户界面：Web 前端，React 架构;
- 后端实现：Spring;
- 后端管理：Maven;
- 统一编程环境：Visual Studio Code (Mac: XCode);
- 项目版本管理：GitHub;
- 建模工具：Rational Rose;
- 数据库：Mongo DB。

## 3 系统特性

### 3.1 功能概览

表 4：用例概览表

序号	用例名	优先级	负责人
0	用户账号注册/登陆	高	全组
1	商品增删/编辑	中	杨文曦
2	商品检索/排序	中	张佳玲
3	用户聊天	低	丁维德
4	订单生成/处理	高	杜培栋
5	订单评价	中	朱重熙
6	用户信息管理	高	Karen

### 3.2 功能详述

#### 3.2.1 用户账号注册/登陆（全组）

表 5：用户账号注册/登陆 用例详述

**用例：**用户账号注册/登陆——利用数据库实现用户账号注册/登陆功能。

**范围：**Platform 二手书交易平台。

**优先级：**高优先级；

**何时可用：**第一个增量；

**使用频率：**高等频度；

**主要参与者：**所有用户；

**涉众及其关注点：**

- 所有用户：需要；

**前置条件：**无；

**后置条件：**将用户注册信息存至数据库方便管理；

**主成功场景：**

场景 1：用户账号注册与登陆

1. 用户点击注册新帐户按钮；
2. 系统开始一次注册操作；
3. 用户填写用户名，邮箱、手机号、密码并确认密码；
4. 系统检测，若格式正确且无重复即注册成功；
5. 用户在登陆界面输入用户名和密码，点击登陆；
6. 系统检测，若与数据库信息对应则登陆成功。

**扩展：**

## 场景 2：注册失败

1. 用户点击注册新帐户按钮；
2. 系统开始一次注册操作；
3. 用户填写用户名，邮箱、手机号、密码并确认密码；
4. 系统检测，格式不正确或已被注册；
5. 显示注册失败提示，并指出失败原因。

## 场景 2：登陆失败

1. 用户在登陆界面输入用户名和密码，点击登陆；
2. 系统检测到与数据库信息部对应；
3. 显示登陆失败提示，并指出失败原因。

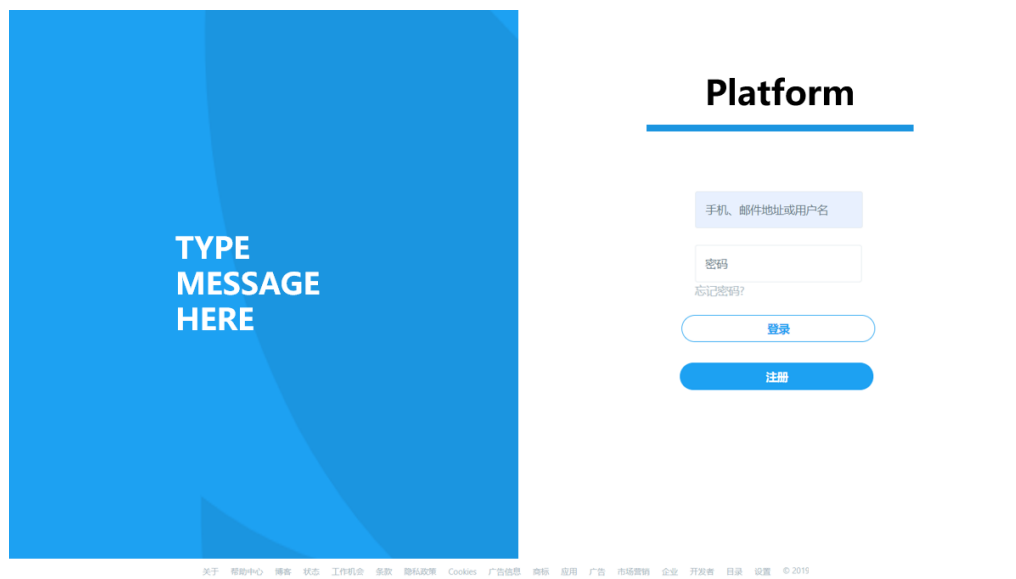


图 1 界面原型

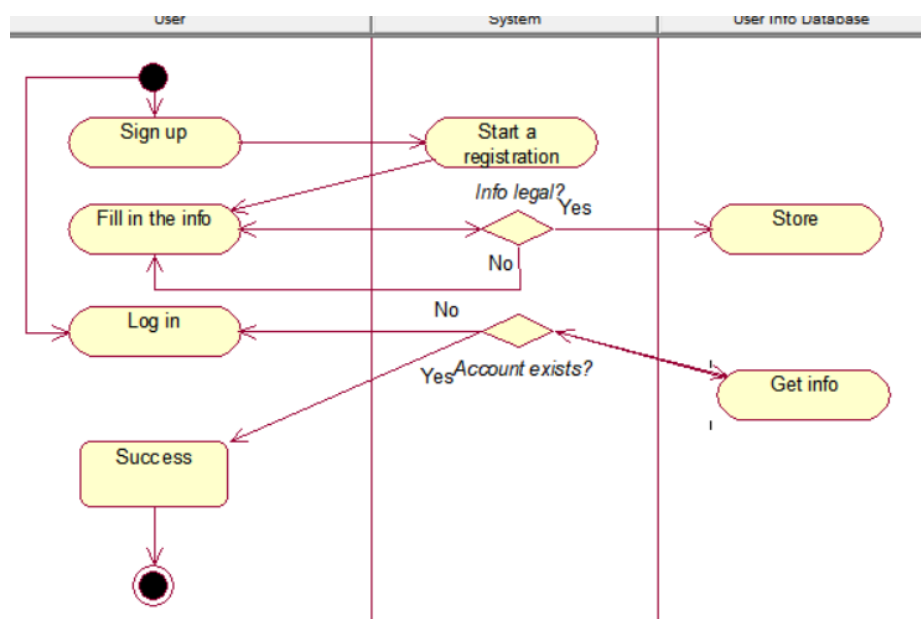


图 2：UML 活动图

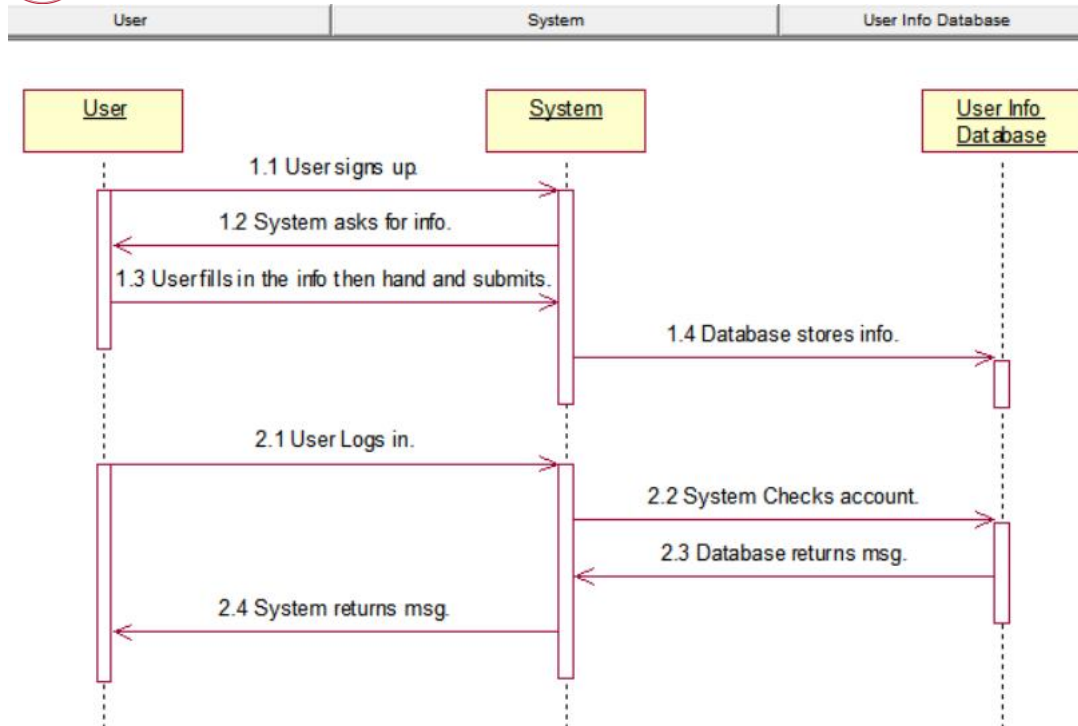


图 3：系统顺序图

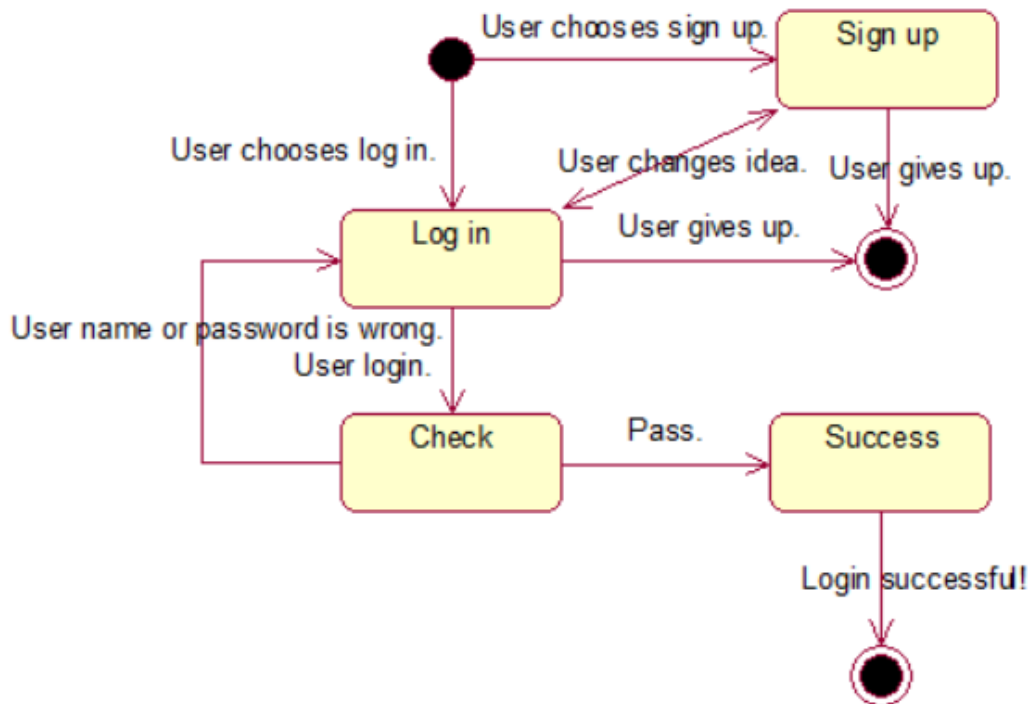


图 4：状态机图



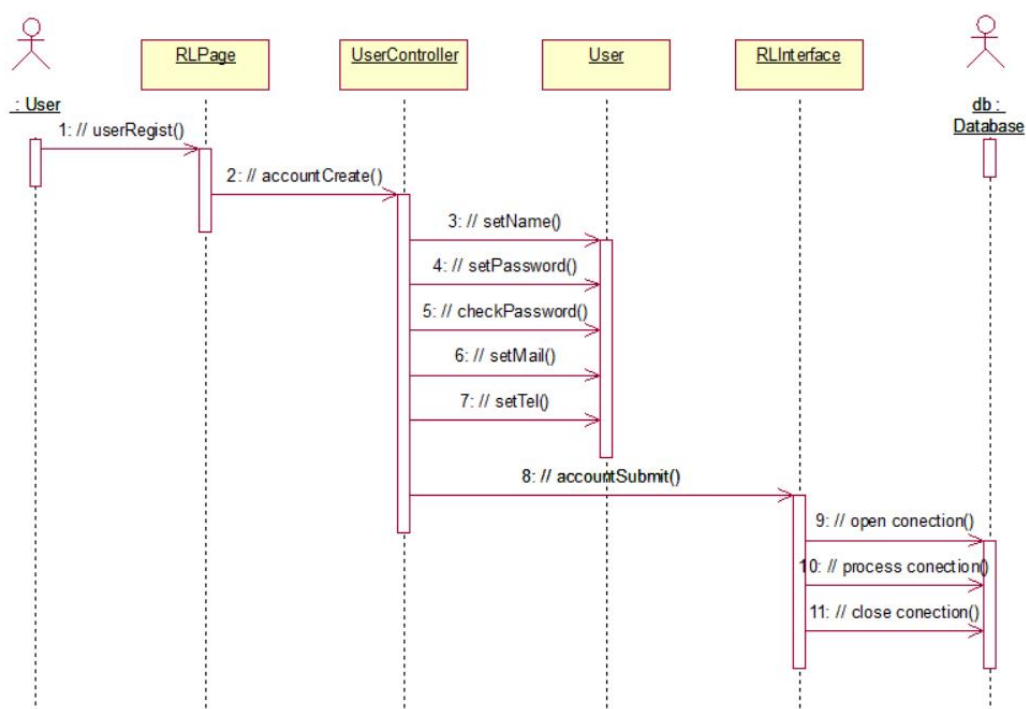


图 5：UML 顺序图

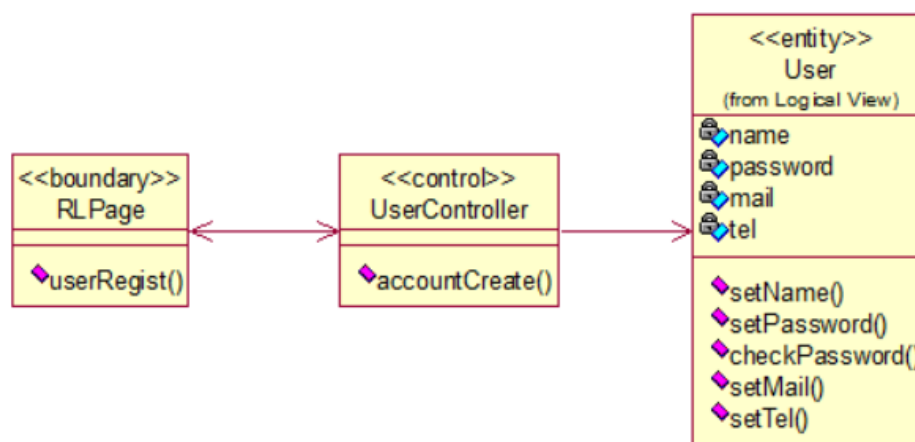


图 6：类图

### 3.2.2 商品增删/编辑（杨文曦）

表 6：商品增删/编辑 用例详述

用例：商品增删/编辑——利用数据库实现管理（上架/下架/更改/删除）商品。

范围：Platform 二手书交易平台。

优先级：中等优先级

何时可用：第一个增量

使用频率：中等频度

**主要参与者：**卖家

**涉众及其关注点：**

- 卖家：希望能够便捷地上架/下架/更改/删除商品，并且能展示商品的关键信息
- 买家：希望能够清晰及时地了解到商品的关键信息。

**前置条件：**卖家必须登陆账号并完成资格认证。

**后置条件：**存储更新后的商品信息以便检索。

**主成功场景：**

场景 1：上架商品

1. 卖家点击上架商品按钮；
2. 系统开始一次上架操作；
3. 卖家选择图书分类；
4. 卖家填写商品信息（名称、作者、照片、品相、描述、ISBN 等）并提交；
5. 系统审核后记录信息，将信息记入商品信息库；
6. 显示上架成功页面，上架成功。

场景 2：编辑商品信息

1. 卖家点击编辑商品信息按钮；
2. 系统开始一次编辑操作；
3. 卖家更改商品信息（有一些信息需要锁定，无法更改）；
4. 系统审核后更新信息，并更新商品信息库；
5. 显示编辑成功页面，编辑完成。

场景 3：下架商品

1. 卖家点击下架商品信息按钮；
2. 显示确认提示信息；
3. 系统更新商品信息库；
4. 显示下架成功页面，下架完成。

**扩展：**

场景 4：必填信息空缺

1. 卖家点击上架商品按钮；
2. 系统开始一次上架操作；
3. 卖家选择图书分类；
4. 卖家填写商品信息并提交，但必填信息有空缺；
5. 系统审核后发现必填信息空缺，提示报错；
6. 返回商品信息填写页面。

## Platform

消息 购物车 我的订单 个人中心 卖家中心 客服 手机版

常用菜单

- 卖家中心首页
- 订单管理
  - 订单管理
  - 合并订单
  - 订单数据统计
- 商品添加
  - 添加商品
  - 扫码上书
  - ISBN批量上传
  - 批量上传商品
  - 添加新书书条目
- 商品管理
  - 草稿箱
  - 管理未售商品
  - 管理推荐商品
  - 管理本店商品分类

图书 > 普通图书/民国旧书 >

\* 书名  0 / 200字

推荐语  0 / 200字

ISBN

\* 作者

\* 出版社

\* 出版时间 ☒ 填写时间 ☐ 选择年代  
 年  月  日 第  版

印刷时间 年  月  日 第  次印刷 网上

印数  千册

\* 装帧 ☐ 平装 ☐ 精装 ☐ 软精装 ☐ 线装 ☐ 其他

开本  请选择

纸张 ☐ 胶版纸 ☐ 铜版纸 ☐ 轻型纸 ☐ 纯质纸 ☐ 其他

页数  页

字数  千字

定价  元

关于 帮助中心 博客 状态 工作机会 多数 隐私政策 Cookies 广告信息 商标 应用 广告 市场营销 企业 开发者 目录 设置 © 2015

图 7：界面原型

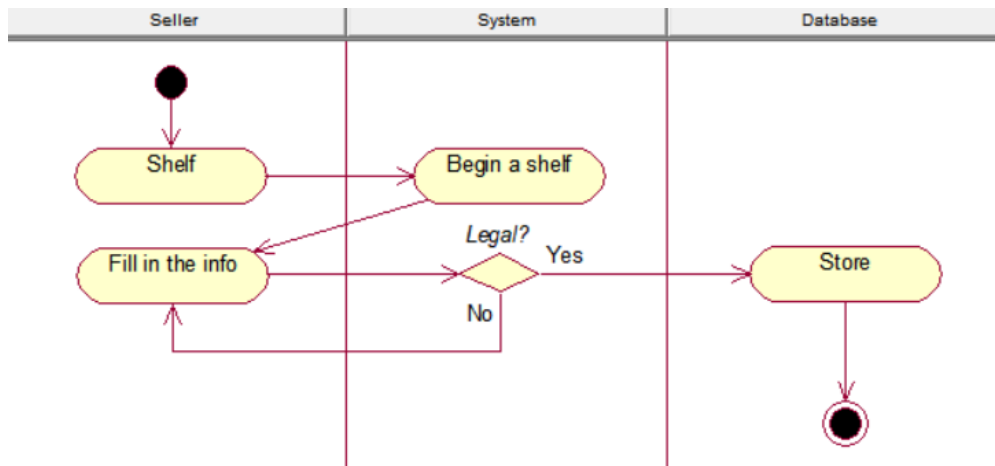


图 8：UML 活动图

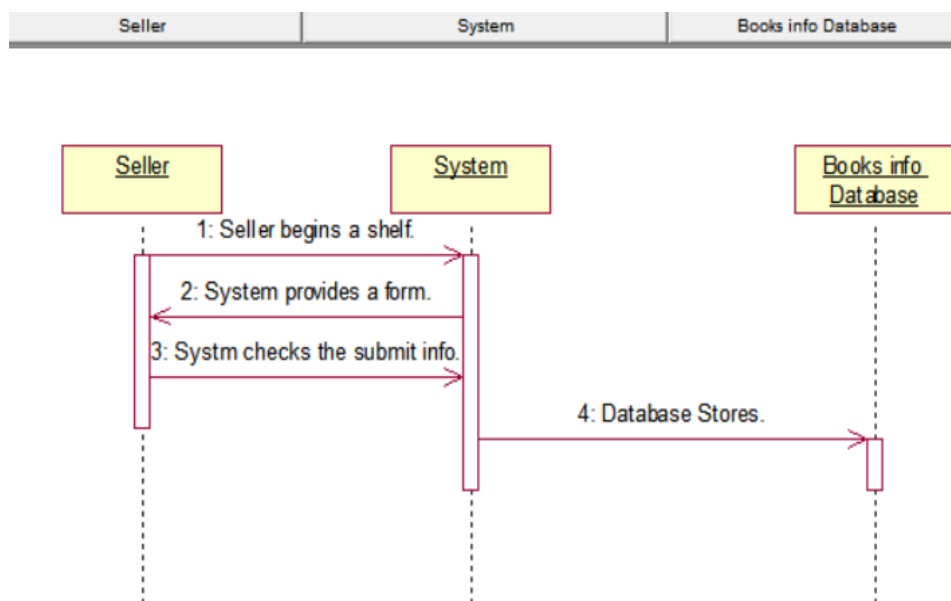


图 9：系统顺序图

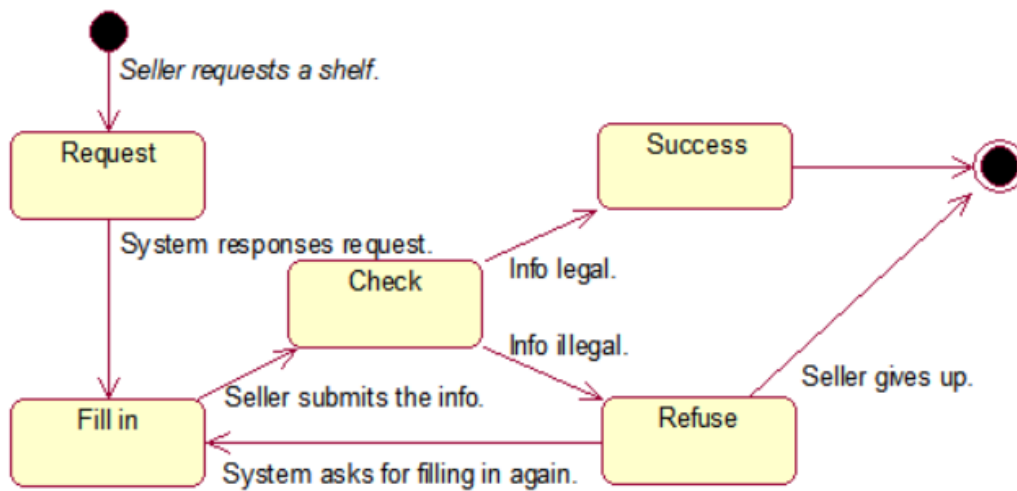


图 10：状态机图

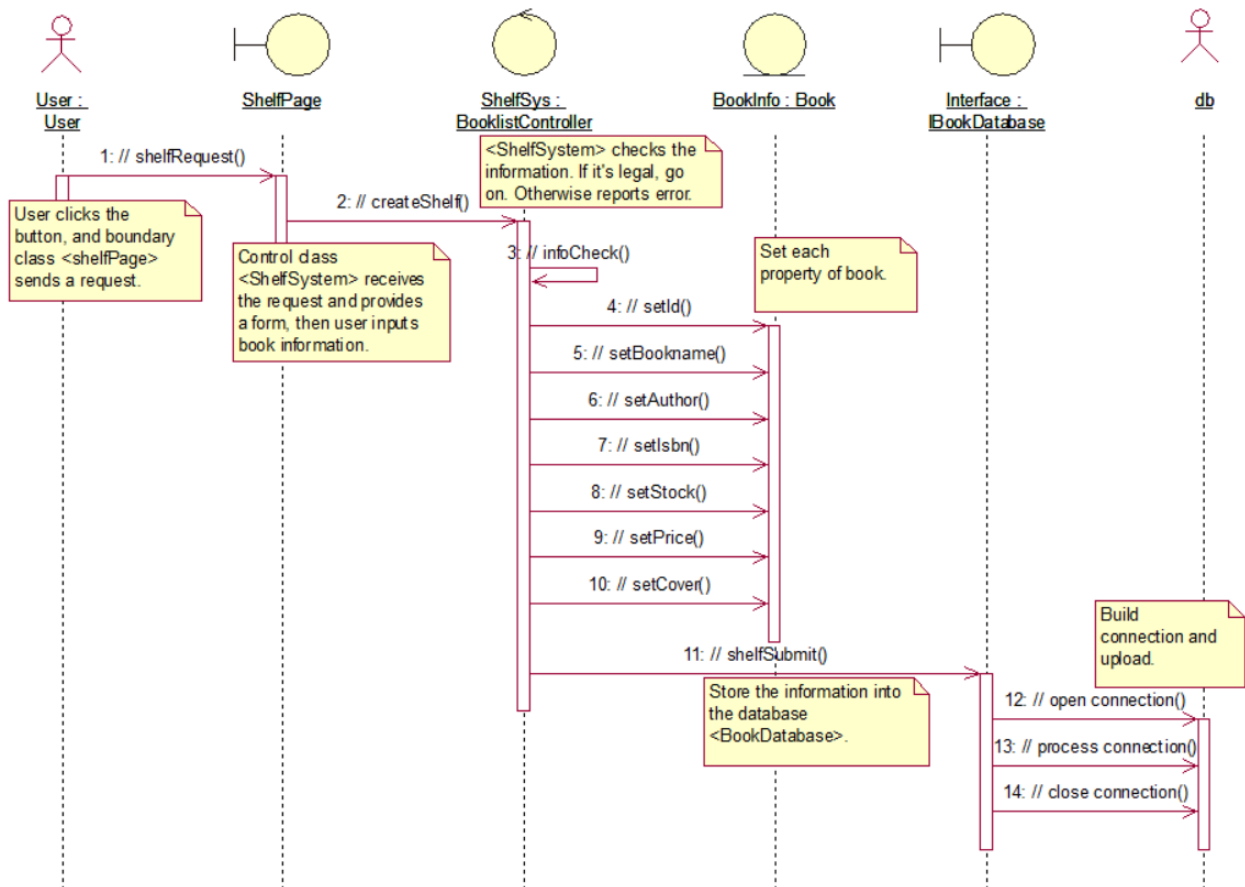


图 11：UML 顺序图

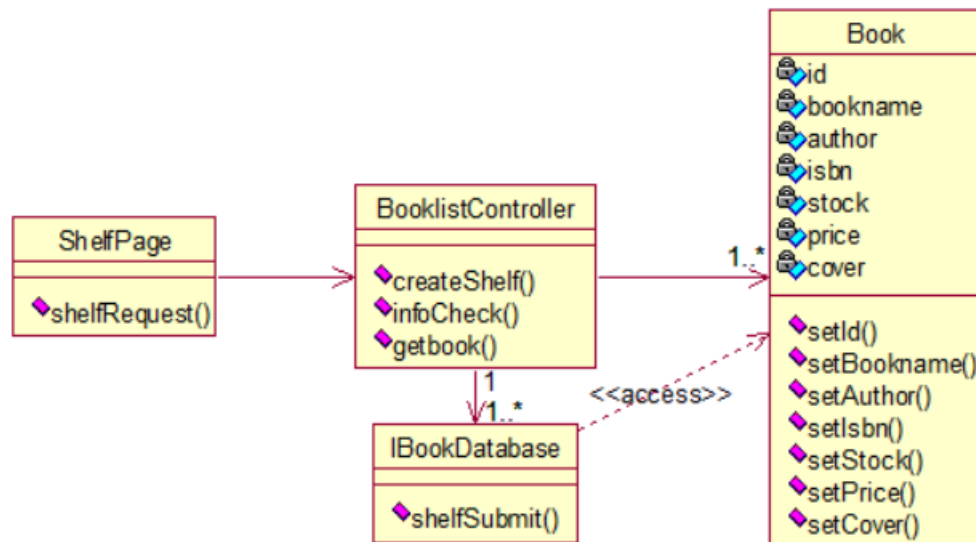


图 12：类图

### 3.2.3 商品检索/排序 （张佳玲）

表 7：查找书籍 用例详述

用例：查找书籍
范围：Platform 二手书交易平台。
优先级：中等优先级
何时可用：第一个增量
使用频率：中等频度
主要参与者：买家
涉众及其关注点：
-买家：希望能快速的找到自己想要的书籍
-卖家：希望自己上架的书籍能被所需要它的买家搜索到
前置条件：买家必须登陆账号并完成资格认证
后置条件：交易完成后在书籍信息库中删除该书籍的信息
主成功场景：
1.买家登陆，并输入搜索信息
2.系统从索引库中查找相关信息（书名、作者名、课程名称等等）
3.系统根据索引信息从数据库中调出相关的书籍信息，并展现给买家
4.买家查看书籍并购买

请输入搜索信息（书本名称、作者名称或课程）

搜索

图 13：用户界面原型

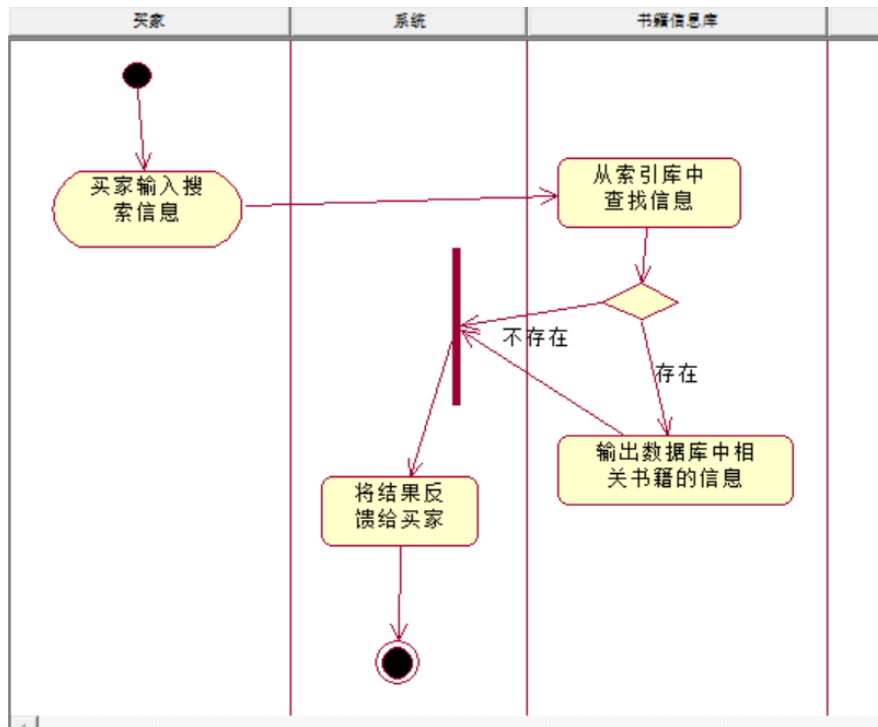


图 14: UML 活动图

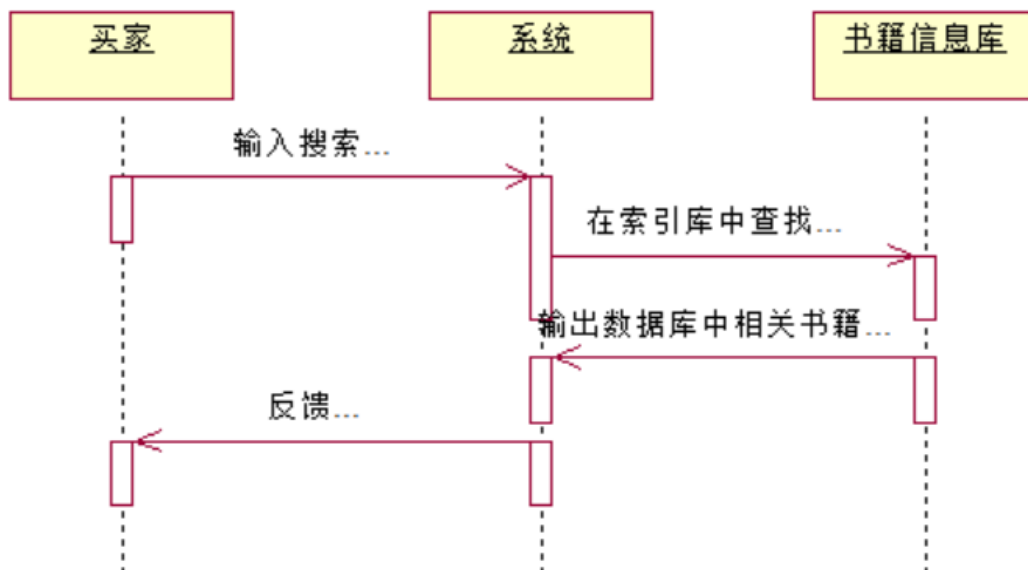


图 15: 系统顺序图

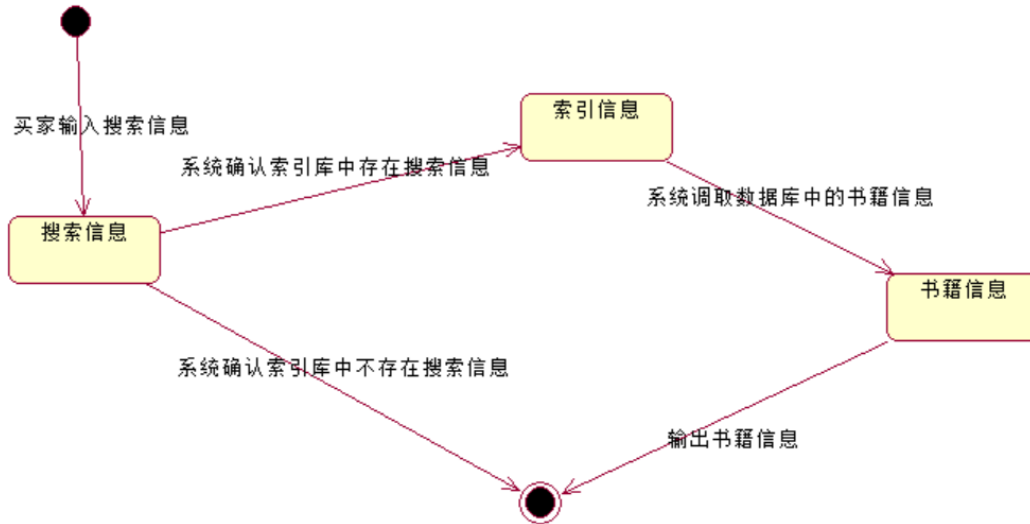


图 16：状态机图

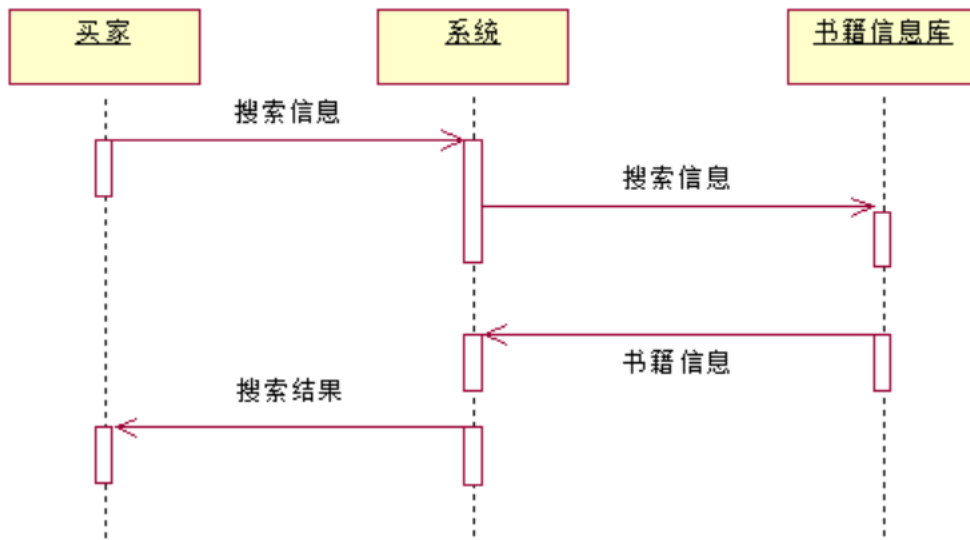


图 17：UML 顺序图

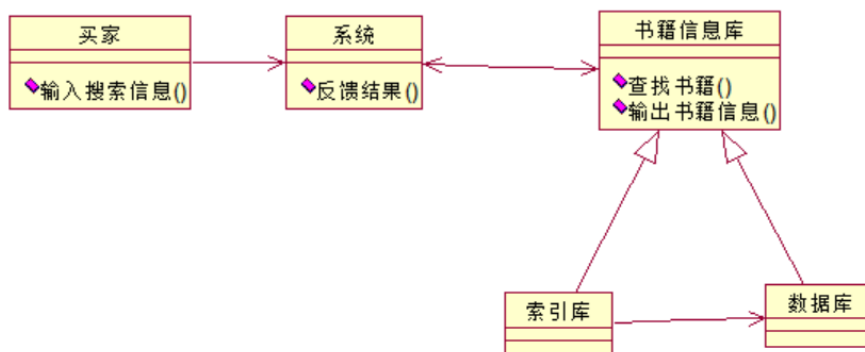


图 18：类图

### 3.2.4 用户聊天 （丁维德）

表 8：用户聊天 用例详述

<p><b>用例：</b> 在线聊天</p> <p><b>范围：</b> Platform 二手书交易平台。</p> <p><b>优先级：</b> 低等</p> <p><b>使用频率：</b> 中等</p> <p><b>主要参与者：</b> 买家与卖家</p> <p><b>关注点：</b></p> <p>互相沟通，交货</p> <p><b>前置条件：</b> 用户必须登录成功</p> <p><b>主成功场景：</b></p> <p>1 买家如果对产品有兴趣，想要更进一步了解此物品，可以透过此用例来与卖家进行沟通</p> <p>2. 如果系统内有新的订单生成如果系统内有新的订单生成，系统将透过此应用自动发消息给买家，从而进行沟通。</p>
--



图 18：界面原型

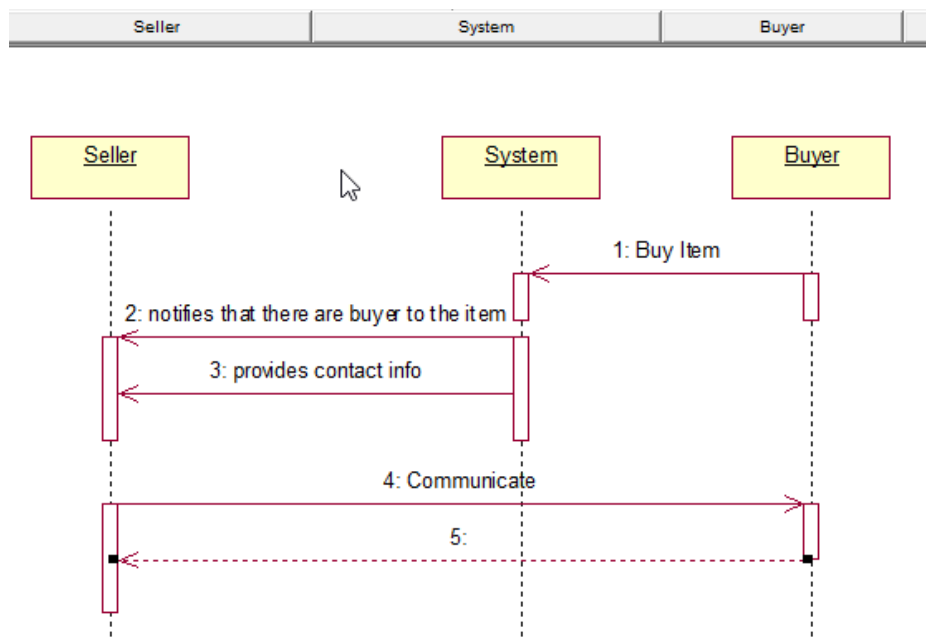


图 19：UML 活动图



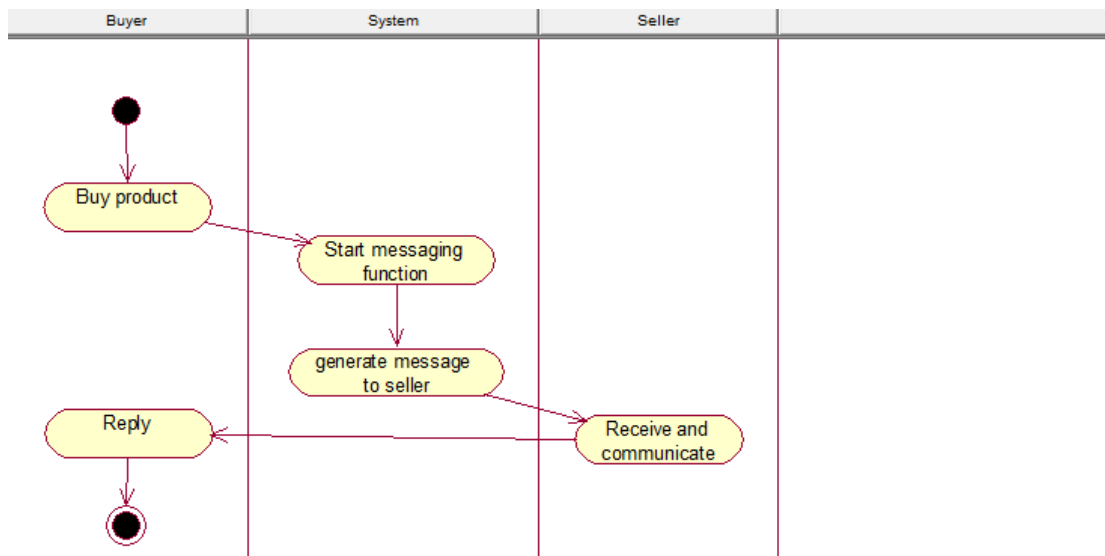


图 20：系统顺序图

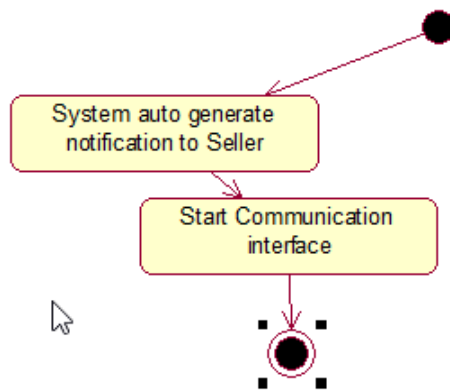


图 21：状态机图

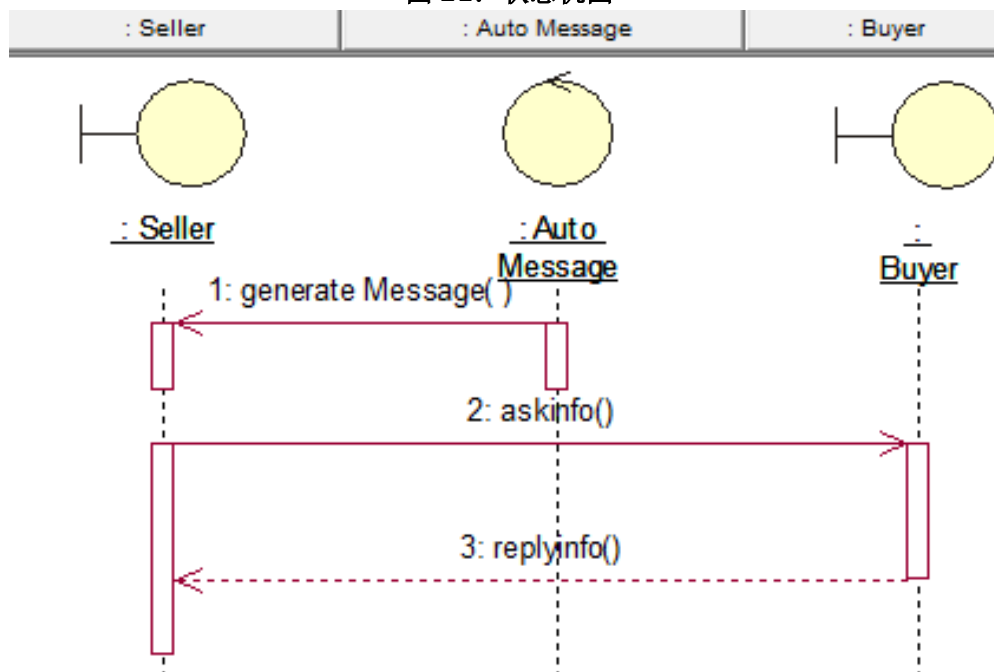


图 22：UML 顺序图

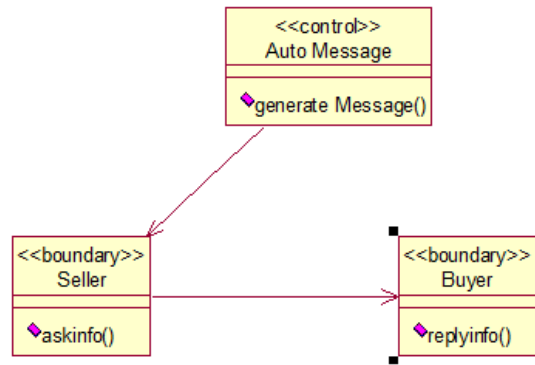


图 23：类图

### 3.2.5 订单生成/处理 （杜培栋）

表 9：订单生成/处理 用例详述

<p>用例：订单生成/处理</p> <p>范围：Platform 二手书交易平台。</p> <p>优先级：高等优先级</p> <p>何时可用：第一个增量</p> <p>使用频率：高等频度</p> <p>主要参与者：买家、卖家、书籍信息库</p> <p>前置条件：买家和卖家在登录状态下操作。</p> <p>后置条件：无</p> <p>主成功场景：</p> <ol style="list-style-type: none"> <li>1.买家填写订单并将其提交至系统；</li> <li>2.系统对订单进行检查，随后将其存入书籍信息库并提示卖家有新的订单；</li> <li>3.卖家接收到新的订单后与买家进行联系完成交易；</li> <li>4.买家向系统确认交易完成。</li> </ol>
---

书籍名称

卖家信息

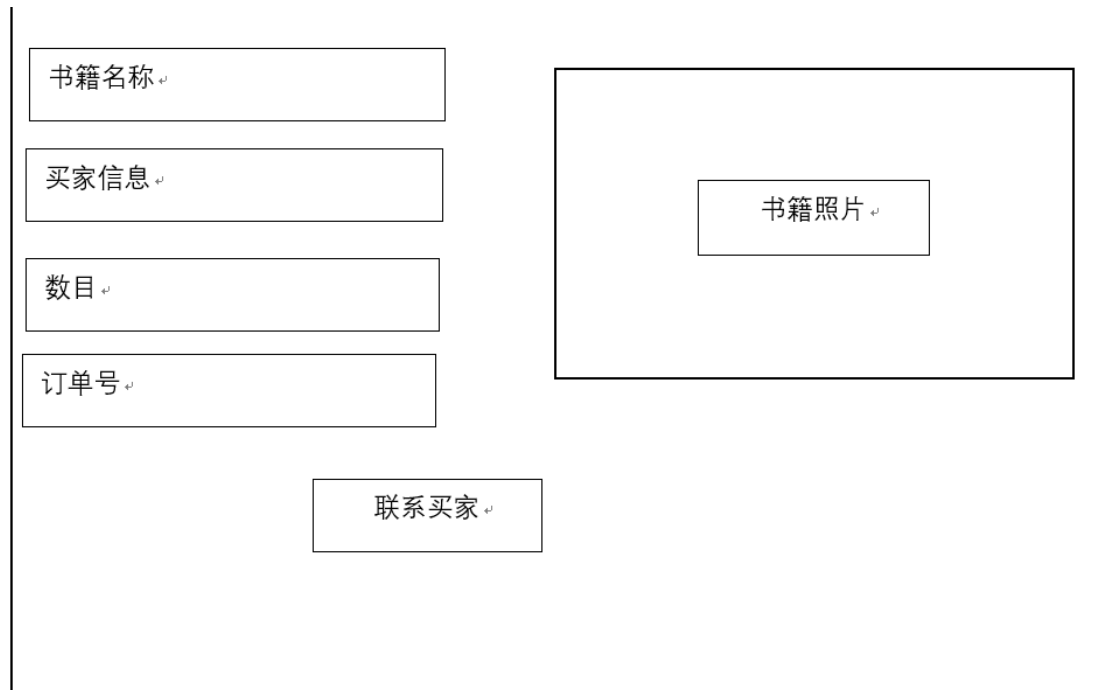
数目

订单号

确认订单

联系卖家

书籍照片



书籍名称

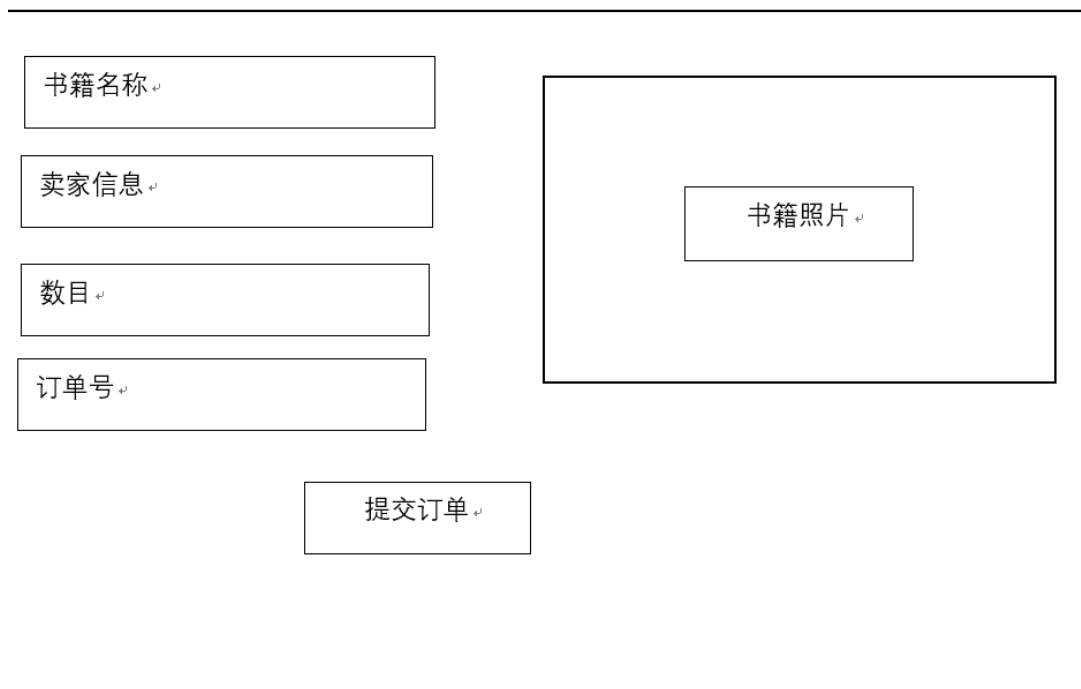
买家信息

数目

订单号

书籍照片

联系买家



书籍名称

卖家信息

数目

订单号

书籍照片

提交订单

图 25：订单生成/处理用户界面原型

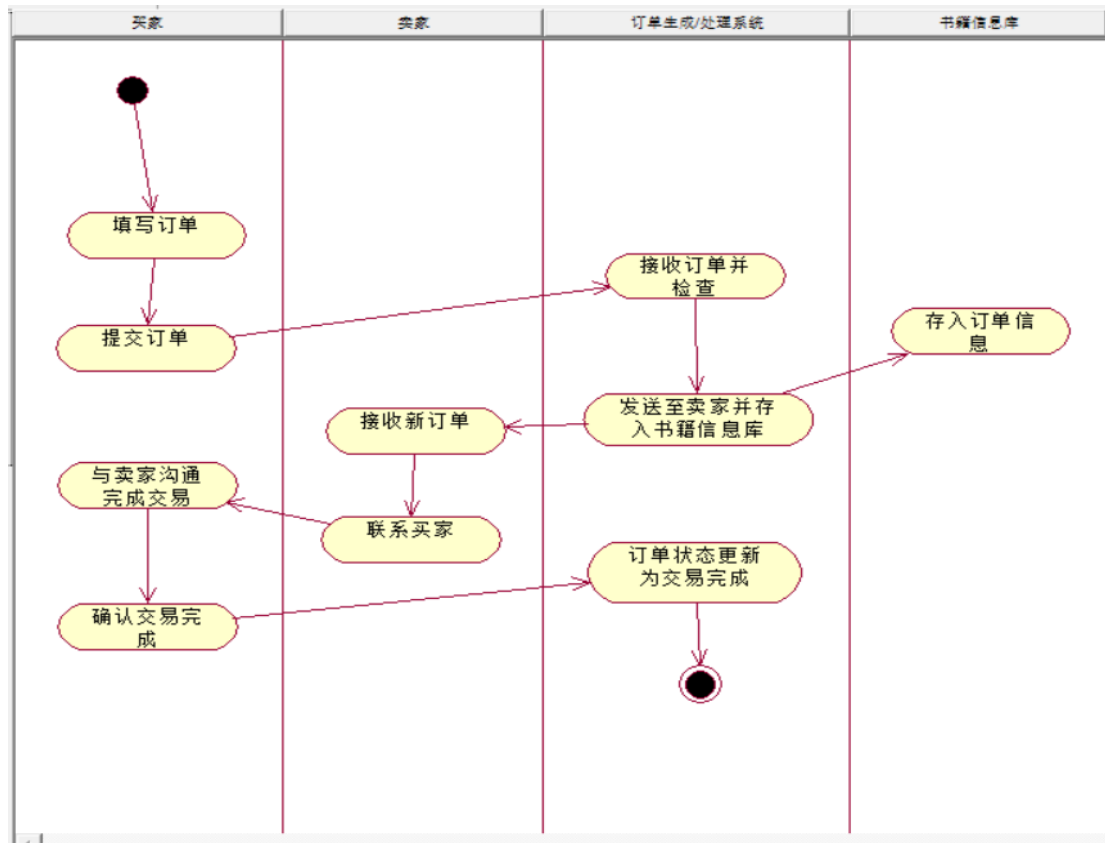


图 26：订单生成/处理 UML 活动图

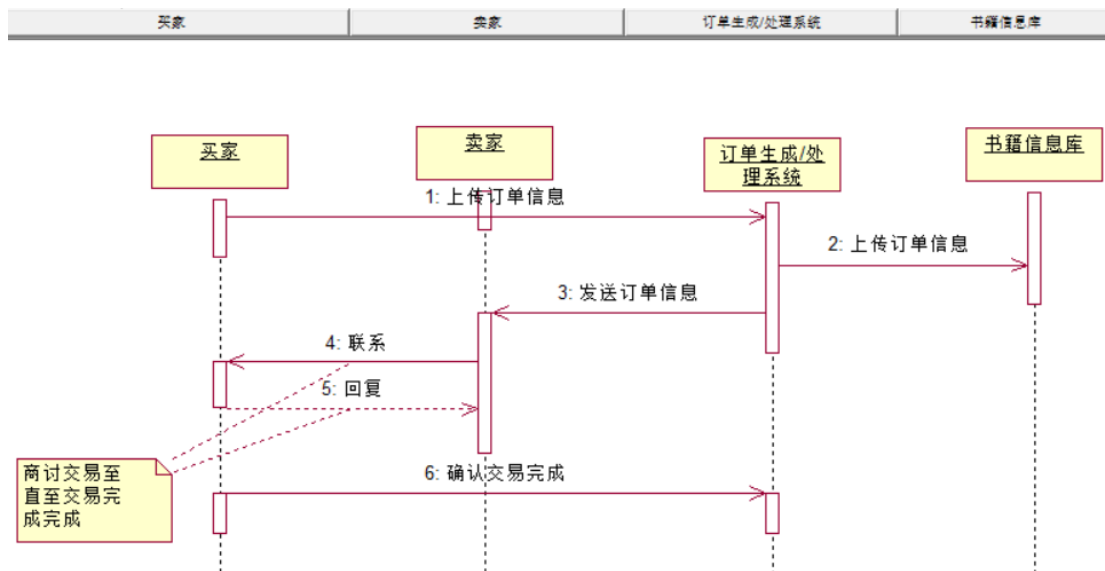


图 27：订单生成/处理系统顺序图

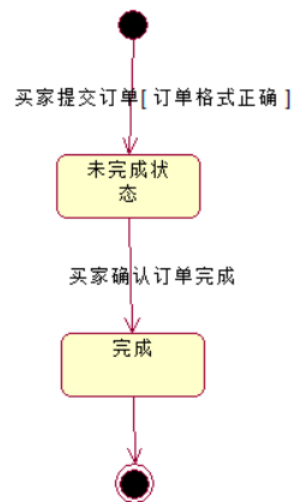


图 28：订单生成/处理状态机图

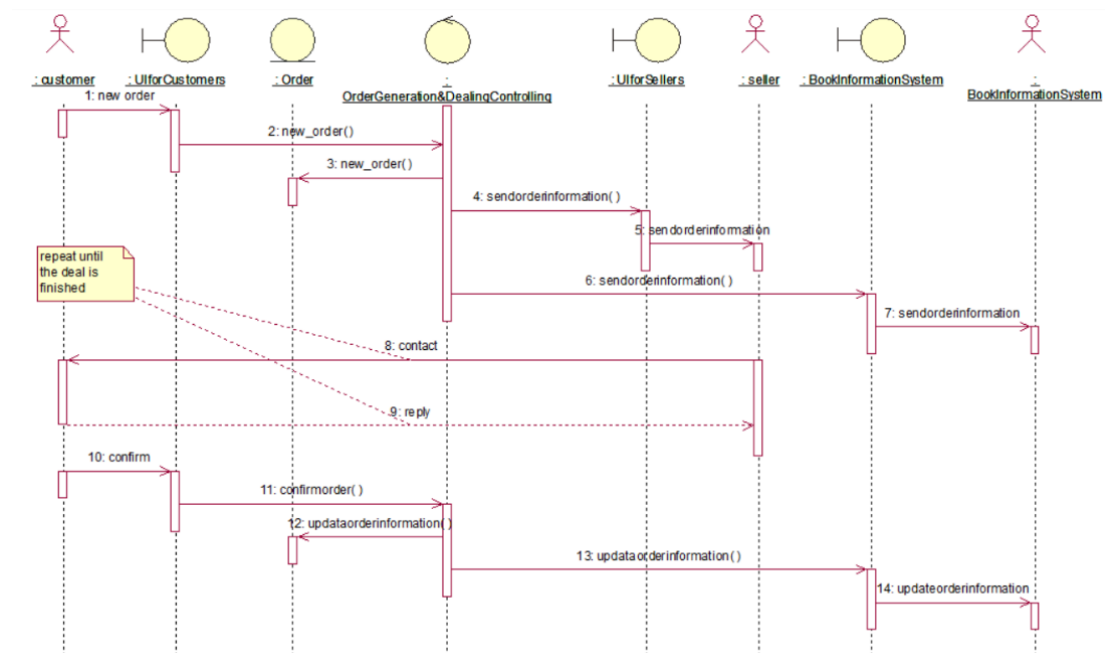


图 29：订单生成/处理 UML 顺序图

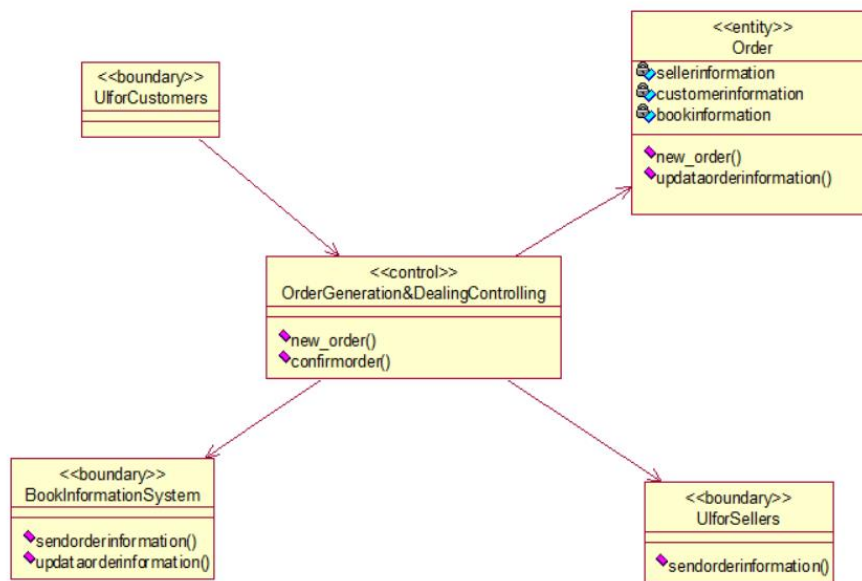


图 30：订单生成/处理类图

### 3.2.6 订单评价（朱重熙）

表 10：订单评价 用例详述

<p><b>用例：</b> 订单评价</p> <p><b>简单描述：</b> 用户对已完成的订单作出评价</p> <p><b>主参与者：</b> 买家、卖家</p> <p><b>副参与者：</b> 数据库</p> <p><b>前置条件：</b> 订单已完成且用户有评价需要</p> <p><b>主流：</b></p> <ol style="list-style-type: none"> <li>1. 用户打开本网站</li> <li>2. 用户进入评价界面</li> <li>3. 在评价界面选择所给星级</li> <li>4. 在评价栏输入评价内容</li> <li>5. 点击确定，发表评价</li> </ol> <p><b>后置条件：</b> 用户评价成功</p> <p><b>附加流：</b> 数据库添加失败时提示错误原因并询问是否重新评价</p>
---

1. 确认订单信息

2. 付款

3. 确认收货

4. 评价

评价须知

店铺信息

某某的书店

被评卖家: 某某


累计得到评价: ★★★★★ 4.9分

店铺动态评分

☐ 全部匿名评论

宝贝

宝贝与描述相符(打分匿名)



95新-《唐诗三百首》火速发货

4分

★★★★☆

(此处输入评价)

还可以输入 492 字

收起评论

公开评论

店铺动态评分

卖家的服务态度 ★★★★★ 4分 - 卖家服务挺好的, 沟通挺顺畅的, 总体满意

卖家发货的速度 ★★★★★ 4分 - 卖家发货还算及时

发表评论

给买家评分

☐ 全部匿名评论



某买家

4分

★★★★☆

(此处输入评价)

还可以输入 492 字

收起评论

公开评论

发表评论

图 31: 界面原型

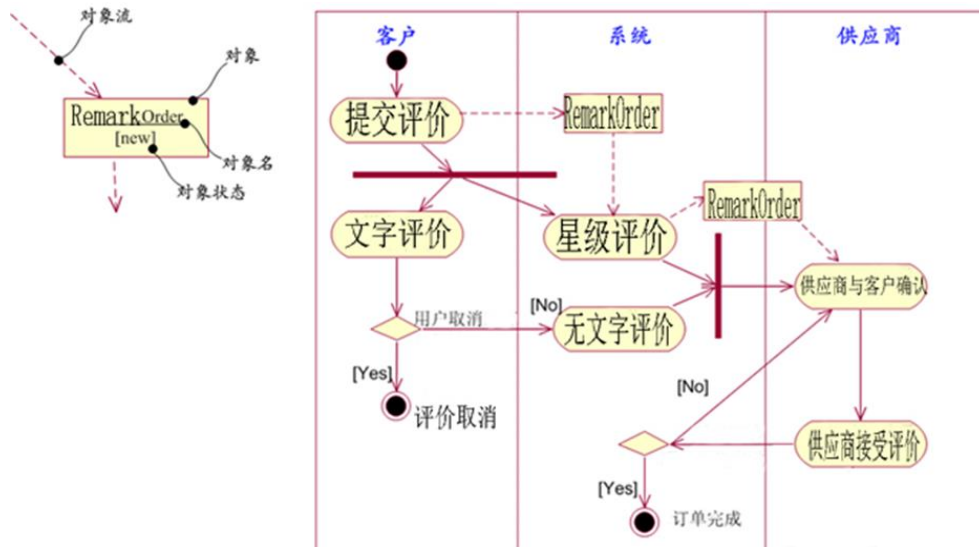


图 32: UML 活动图

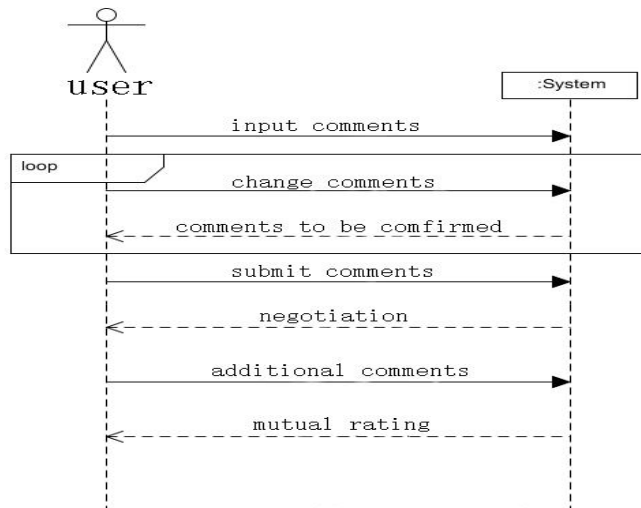


图 33: 系统顺序图

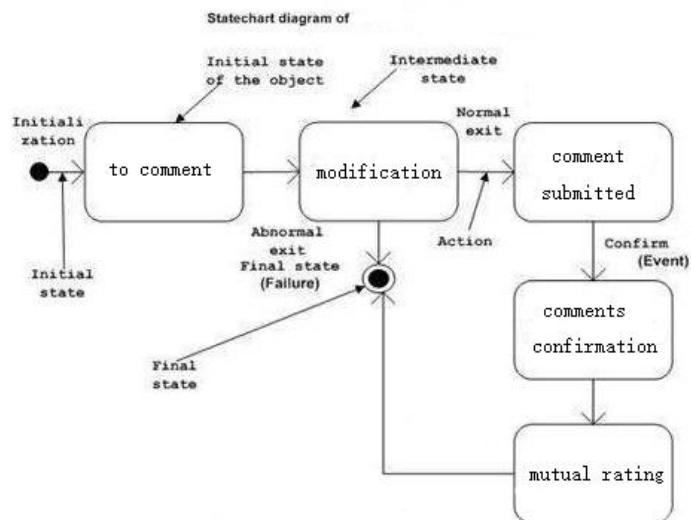


图 34: 状态机图



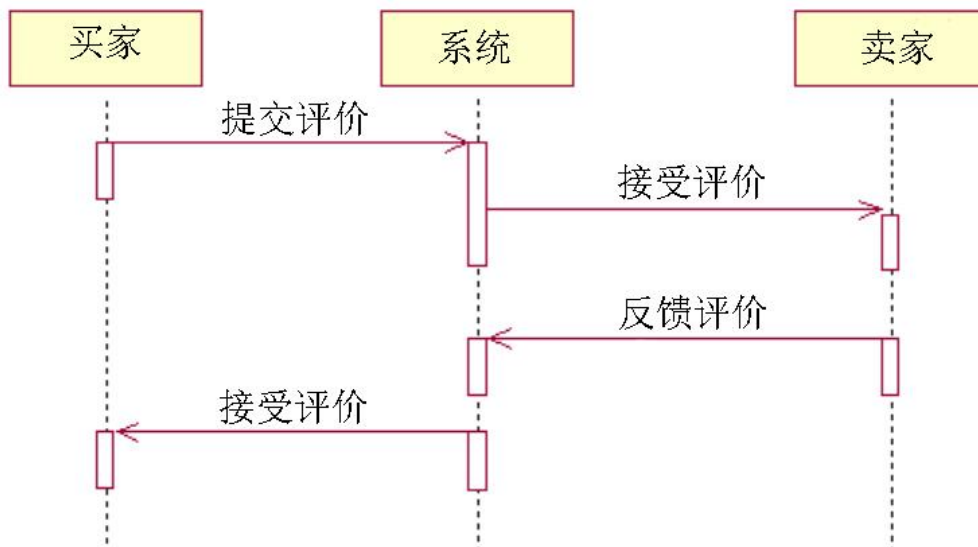


图 35: UML 顺序图

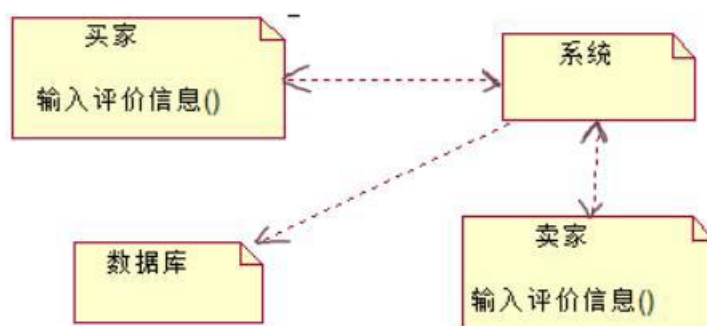


图 36: 类图

### 3.2.7 用户信息管理 (Karen)

表 11: 用户信息管理 用例详述

<p><b>用例:</b> 用户信息管理——实现用户封禁。</p> <p><b>范围:</b> Platform 二手书交易平台。</p> <p><b>优先级:</b> 低等优先级</p> <p><b>何时可用:</b> 第二个增量</p> <p><b>使用频率:</b> 低等频度</p> <p><b>主要参与者:</b> 所有用户</p> <p><b>涉众及其关注点:</b></p> <ul style="list-style-type: none"> <li>- 所有用户: 对信用不良或有其他违法行为的用户实行封禁, 净化平台环境。</li> </ul> <p><b>前置条件:</b> 用户必须注册。</p>
--

后置条件：存储封禁账号具体信息。

主成功场景：

1. 接到用户举报；
2. 系统核实，封禁；
3. 被封禁用户有异议可以上诉，再次核实。

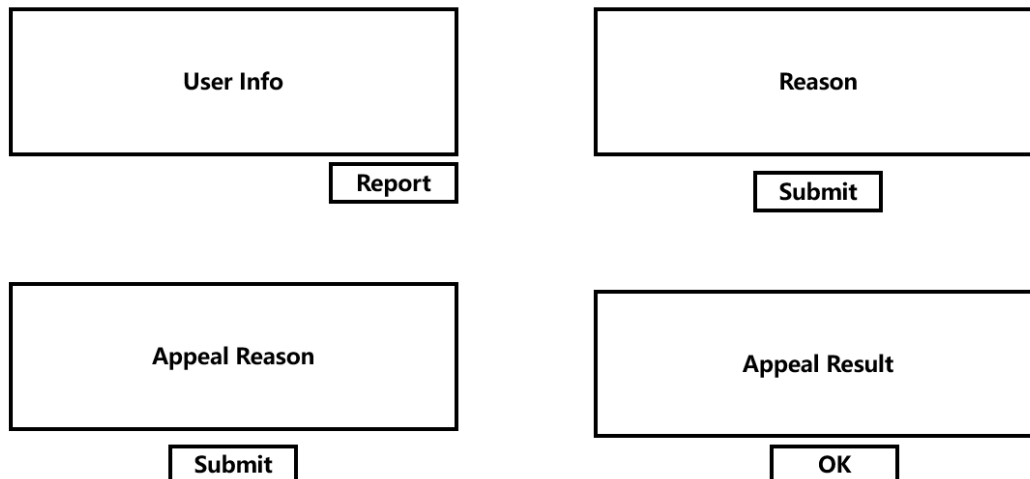


图 31：界面原型

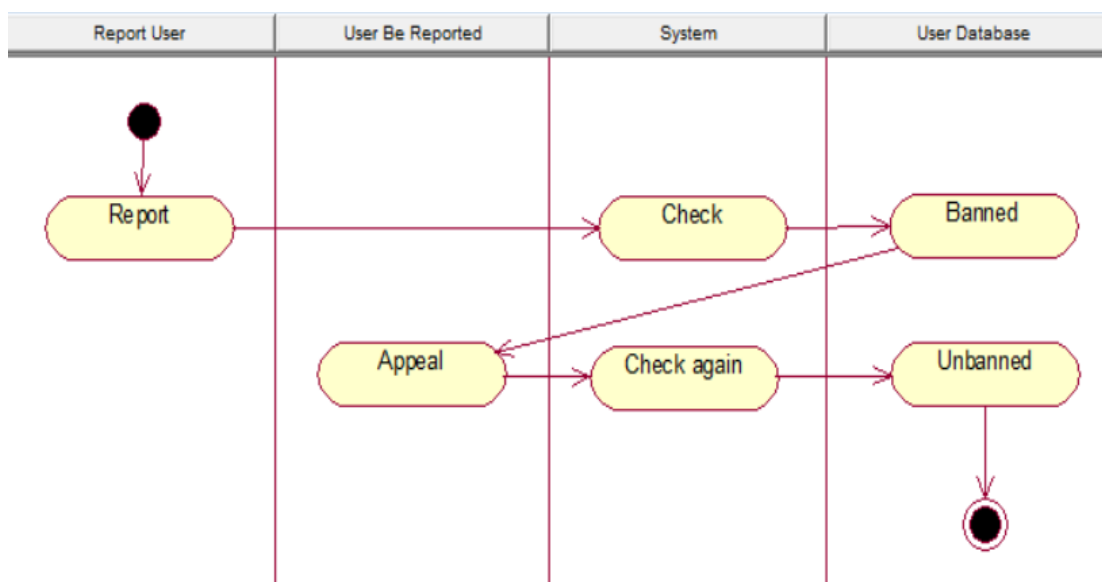


图 32：UML 活动图

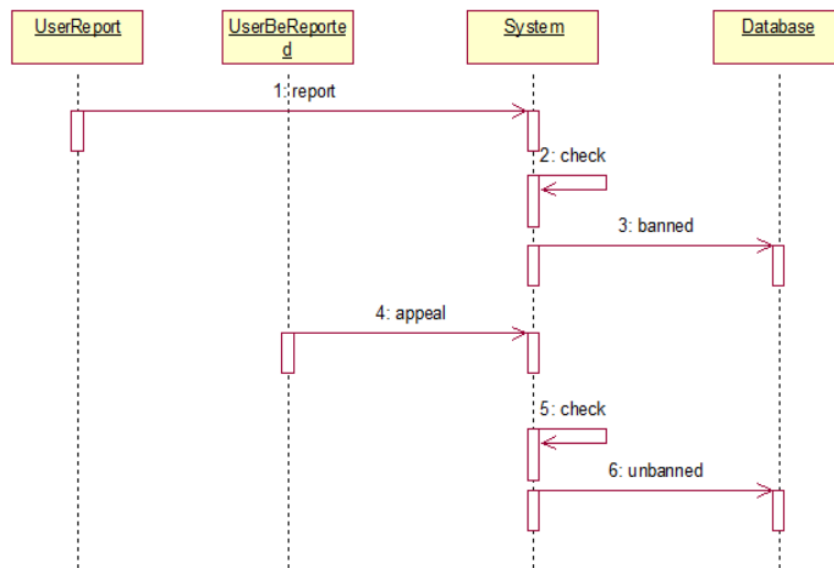


图 33: 系统顺序图

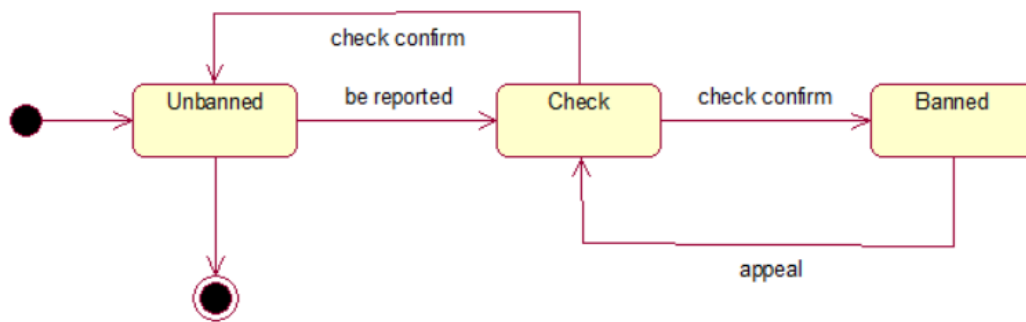


图 34: 状态机图

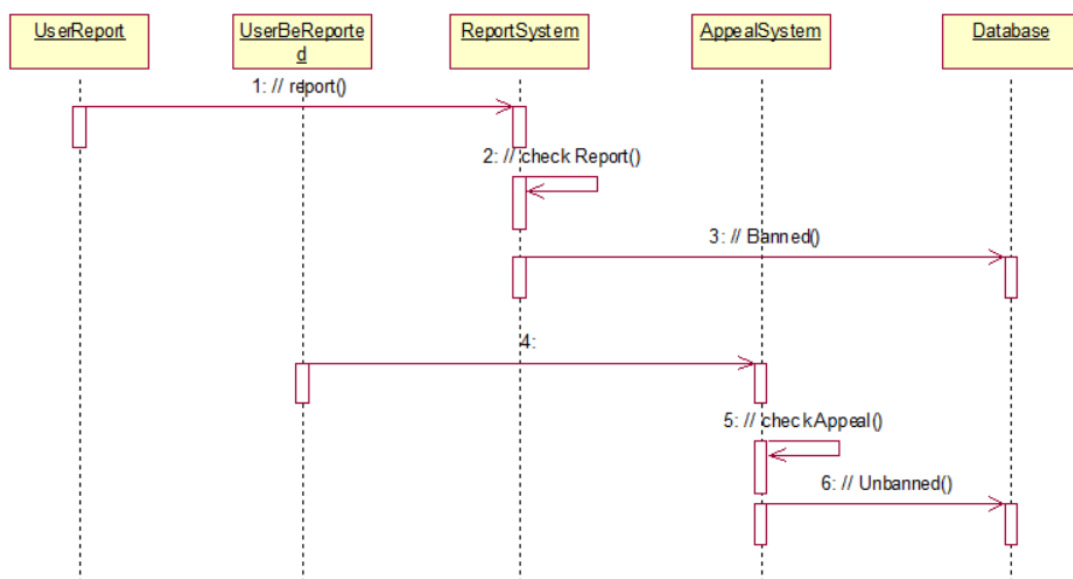


图 35: UML 顺序图

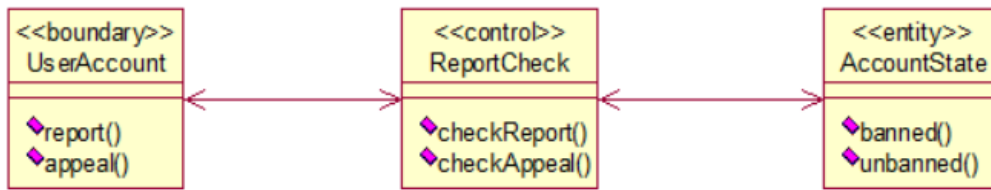


图 36：类图

## 4 外部接口需求

### 4.1 用户界面



图 37 用户界面原型

### 4.2 硬件接口

无外部硬件接口。

### 4.3 软件接口

- 使用 Facebook 开发的 react 框架；
- 使用 Mongo DB 数据库。

### 4.4 通信接口

无外部通信接口。

## 5 其他非功能需求

### 5.1 性能需求

- 足够的终端数支持（本课程迭代时不考虑）；
- 足够的并行操作用户数支持（本课程迭代时不考虑）；
- 系统响应特征时间应在 0.1s 以内；
- 动态数值需求（本课程迭代时不考虑）。

### 5.2 安全需求

强制访问控制。

### 5.3 软件质量属性

友好性：操作界面和报错信息需从用户角度考虑；

安全性：本课程迭代时不考虑；

可移植性：软件需使用 react 框架和 npm 管理系统，前端采用 WebPack 打包管理，后端采用 Maven 打包管理。数据库间的转换暂不考虑；

可扩展性：顶端应有抽象接口设计的类层次结构，代码结构应有充分逻辑且可塑性较强；

可修改性：代码应有较强的可读性与充分的注释以便后续修改。

## 附录

### 附录 A 业务实体

- 用户账号 (User)
- 书籍信息 (Book)
- 订单 (Order, OrderItem)
- 购物车 (Cart)

### 附录 B 分析模型 (UML 模型)

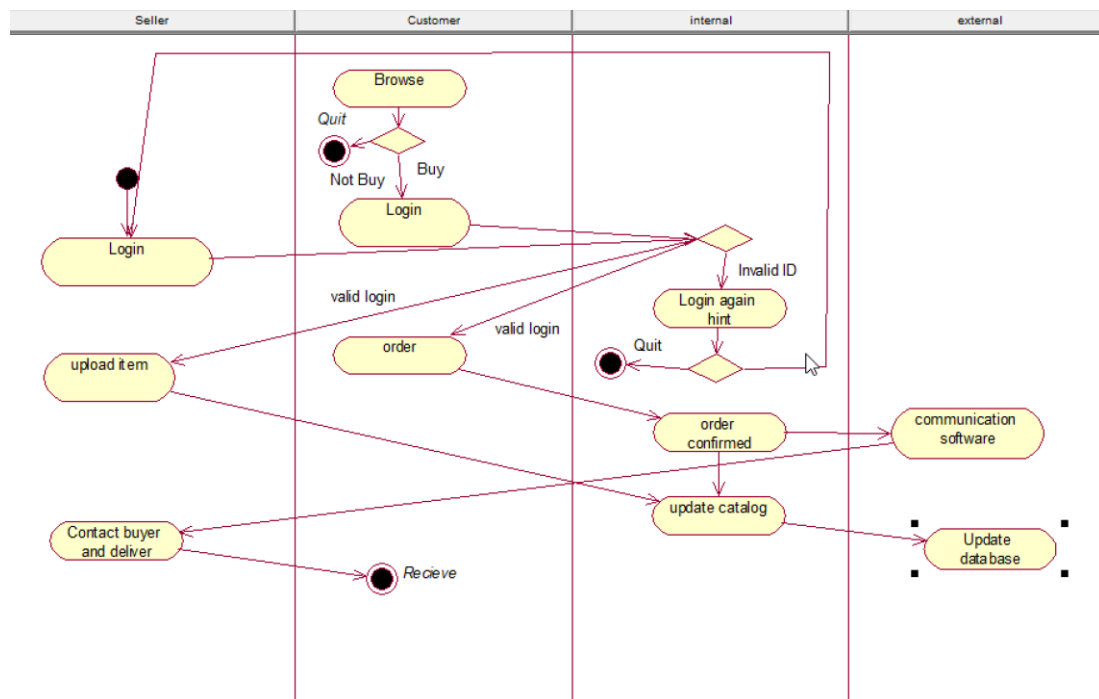


图 38 带泳道的核心业务 UML 活动图

其余各 UML 图分类归结在 3.2 各部分，有详细标注。