**第二阶段设计\_\_开发文档**

**功能模块：**

**图书管理**：图书入库(图书进库) 图书上架 图书下架 图书信息修改 图书分 类及查询

**购物车：**

**订单管理：**订单信息 订单物流状态 订单发货 订单退货处理等

**会员管理：**会员申请 会员信息(优惠政策或折扣) 会员升级

**财务统计：**当日、当月、当年的进货花费详情。当日、当月、当年总收入以及总利润， 按日、月、年来查询收入信息以及利润信息

**售后板块：**在线客服，评论查看，该商品评分以及星级

**权限划分：**系统管理员 商家 买家

1. 图书信息应包含：图书编号(没一类书对应唯一的一个编号)、图书名称、图书类型、 作者、定价、出版社、出版日期、图片信息等。
2. 图书分类(考虑淘宝的图书分类)：文学、小说、儿童读物、考试/教材/论文、工业/ 农业技术、艺术、社会科学、保健/心理类书籍、历史、励志、政治军事、医学卫生、 计算机/网络、外语/语言文字、法律、旅游、经济、漫画/动漫小说、传记等，具体 分类到时再讨论。
3. 图书入库(相当于进货)应包含： 图书编号(每类图书对应一个编号)、图书名称、图书 类型、进货数量、进货单价、进货总价、进货日期、操作人。入库的操作者应该是 商。
4. 图书上架以及下架：刚入库的图书不能显示在商品目录上，必须经过上架处理才可 以被顾客看到以及购买。
5. 图书查询分为：按图书名查询、按作者查询、按类别查询。
6. 购物车(参考淘宝的购物车)：显示物品信息、可修改数量、选中购买、批量删除等
7. 订单信息：包含订单号、订单日期、当前物流信息、详情物流信息、收货人(姓名及电话)、收货 地址、买家留言、购买的物品、订单总价、实付款。
8. 订单与物品库存：物品的库存量是否减少最终以实际订单为准，而不是以是否加入 购物车为准。
9. 订单退货的问题：当顾客要退单时，默认向后台发送一个退单请求。只有当商家或 系统管理员批准时才能真正退单，这时需要将该订单收回并将数据库中的库存量恢 复。
10. 会员管理：顾客可以选择办理会员业务。会员享受的特权到时候具体讨论。
11. 财务统计：应显示进货花费详情，当日、当月、当年收入以及获得利润详情。还必 须实现按日期查询进货或者收入详情。财务统计的内容只允许查看或者使用其统计 结果，不允许修改，若要修改必须到数据库内修改，所以统计的直接对象可以是直 接访问数据库，进行数据库的查询方面。
12. 在线客服：
13. 评论以及评分：当顾客确认收到货时，才能对该物品进行一次评价(包括评分)，也可 展示图片。在顾客购买该商品时也可以查看已经购买的顾客的评价详情。
14. 权限划分：
15. 系统管理员：
16. 买家：只能查看商品购物页面，商品评价等信息。可购买商品
17. 卖家：可对一些商品信息修改，上下架操作，商品入库等。

**数据库1.0**

图书（图书编号，名称，类型，作者，定价，出版社，出版日期）；

商家（商家id, 姓名，密码，性别，联系电话，店铺名称，店铺所在城市，店铺描述, 店铺创建日期）；

商家拥有的图书库存（图书编号，商家id，数量，售价，图书的描述，图片地址1，图片地址2，图片地址3）；

顾客（顾客id，密码，姓名，性别，出生日期，手机号，）；

交易记录（交易号，商家id，图书编号，顾客id，购买数量，交易日期）；

收货地址（顾客id，收货人，省，市，县，街道，联系电话，邮编）；

评论（评论id, 顾客id，评论信息，评论目标图书编号，评论目标商家id,）

评论的解释：一个顾客可以对指定的图书评论，为了减少开发评论层只有一层，该评论区不可以被其他人回复。

**注意：带下划线的为主键。**

book(**b\_no**,name,type,author,price,publish,publish\_date);

seller(**s\_id**,name, password, gender, birthday, tell\_no，shop\_name, city, introduction, create\_date);

Sb(**b\_no, s\_id**, number, sell\_price,img\_address1,img\_address2,img\_address3);

customer(**c\_id**, password, name, gender, birthday, tell\_no);

transaction(**t\_id**, **s\_id, b\_no, c\_id**,number, date);

address(c\_id, name, province, city, county , street, tell\_no, post\_code);

Comment(**c\_no**, c\_id, comment\_info, b\_no, s\_id );

图书编号：年份+流水号，最大10个字符

商家id : 年份+0+流水号，最大10个字符

顾客id: 2017+1+流水号，最大10个字符

交易号：自增

会员号：20152470431

**关于图书的图片访问问题**：

由于我们不能将图书的图片存到数据库（一种不好的行为），所以我们只能将图书的图片地址存到数据库。

在这里，我们不能把图片地址存到 ’图书’ 实体表中，如果这样做的话，那么不同商家所卖的同一种书用的是同一个图片信息了，那么图片应该有谁来上传，系统管理员？ 或者是商家？ 哪位商家？显然这种方法不太现实。

所以图书的图片地址应该存放到 ’商家拥有的图书库存’ 关联表中，这样的意思是：每个商家所拥有的图书有着自己的一套图片信息。

那么问题又来了：我们如何找到每个商家所拥有的图书的图片地址呢？

解决方案：每当一个新的商家注册成功时，我们应该在后台做一件事：’以刚注册的商家的id作为名称，在图书的图片库内创建一个文件夹，里面全都存放该商家拥有的图书的图片’。

**为了方便测试，我们的项目图书图片库地址统一为：’D:\bookshop\book\_img’**

**以后我们往数据库里存的图片地址应为如下格式：D:\bookshop\book\_img\商家id\图片名**

**关于我们的项目的数据文件存放问题？**

D:\bookshop

**关于店铺图片存放问题：**

为了达到统一，我们项目店铺的图片保存地址统一为：“D:\bookshop\shop\_img”下；

具体格式应为：**D:\bookshop\shop\_img\店铺编号\图片名**

**数据库2.0**

图书（图书编号，名称，类型，作者，定价，出版社，出版日期）；

商家（商家id, 姓名，密码，性别，出生日期，手机号）； //商家id相当于身份证

店铺( 店铺号，店铺名称，所在省，所在市，所在县，街道，邮编，店铺描述，店铺封面图片地址，商家id);

店铺拥有的图书库存（图书编号，店铺号，数量，售价，图书的描述，图片地址1，图片地址2，图片地址3）；

顾客（顾客id，密码，姓名，性别，出生日期，手机号）；

交易记录（交易号，商家id，图书编号，顾客id，购买数量，交易日期）；

顾客收货\_地址（顾客id，收货人，省，市，县，街道，邮编，联系电话）；

地址（地址编号，省，市，县，街道，邮编）；

评论（评论id, 顾客id，评论信息，评论目标图书编号，评论目标店铺id, 评论日期）

**注意：带下划线的为主键。**

book(**b\_no**,name,type,author,price,publish,publish\_date);

seller(**s\_id**,name, password, gender, tell\_no);

Shop(**s\_no**,name,province,city,country,street,post\_code,introduction,img\_address,s\_id);

Sb(**b\_no, s\_no**, number, sell\_price,book\_info,img\_address1,img\_address2,img\_adress3);

customer(**c\_id**, password, name, gender, birthday, tell\_no);

transaction(**t\_id**, **s\_no, b\_no, c\_id**,number, date);

address(c\_id,name, province, city, county , street, post\_code,tell\_no);

Comment(**c\_no**, c\_id, comment\_info, b\_no, s\_id,date );