

# 问题及解决方案汇总

MoriSupa 2016

---

## 模糊

Q1. 对于借还，预约，办证等流程的叙述过于简单

A1. 在面谈期间，我们向前台人员进行了详细的询问。（见 ../records/interview/records\_of\_conference）

Q2. 维护人员对于系统遭到安全性威胁时应当采取的行为未做说明

A2. 询问后，维护人员说不需要考虑。但是，为了系统的可靠性，在需求建模的时候，我们采用了对后端数据库操作进行了封装的设计，即所有相关人员对于数据的操作都要经过这一层封装。封装层对于请求进行过滤和记录。维护人员也可以通过管理界面，监测外界对系统的访问情况。

Q3. 维护人员在提到安全性的时候，说到了数据不可以随意修改和访问，但是没有明确各个角色的权限

A3. 面谈时我们询问了这个问题，但是没有得到十分明确的回复。我们根据安全性和各类信息的使用情况构建了这样的权限规则

- 1) 读者：查看和修改自己的基本信息（包括密码），查看书籍的基本信息和当前状态
- 2) 财务：查看书籍的基本信息和可用性信息（贵重度，损毁度，...），操作书籍的是否报废
- 3) 前台：查看和书籍的基本信息和可用性信息只能当读者提供了身份证明是才能获得对应读者的基本信息（密码除外），当得到读者确认，才可操作对应信息
- 4) 维护：大致上对于所有信息包括系统参数都有查看和操作的权限，然而通过与后台数据库的封装的接口

---

## 冲突

Q1. 读者认为2.0倍的丢失赔偿价格过高，并希望原价赔偿；而财务人员认为原价赔偿不能接受

A1. 经过我们负责人员的调解，双方都接受了1.5倍赔偿的规则。

Q2. 读者认为100元的预付金使得可借书籍的本数过少，而财务人员认为提高初始预付金会使得用户进入成本过高

A2. 我们提出让读者自愿多交预付金的方案，经过财务人员的考虑，最后决定了每次借书时可以添附临时预付金的方案。（见 ../records/contact/）

---

## 不可行

Q1. 实体借书卡会使得图书馆的运营成本进一步提高，并且对于读者来讲，借书卡可能遗失，忘记携带

A1. 我们的方案是读者的注册，只需要身份证和手机（或邮箱）的双认证即可为之生成虚拟的图书馆服务使用凭证（即“虚拟借书卡”）。这样读者每次只需要通过身份证认证便可以轻松享有图书馆的所

有服务，而信息的修改，提醒的通知也只需通过预留的联系方式进行。这种方案降低了成本的同时，增加了对读者的友好度。

Q2.在财务提出了临时预付金的概念之后，发生了一个可能的复杂情况。按照临时预付金对应批次的借阅的书籍，而续借等操作使得同一批借阅的书籍可能不是同时归还，那么临时预付金的退换就出现了问题。这样的设计是不可行的，或者说至少是相当复杂的。

A2.我们认为临时预付金是分片“沉积”在对应的借阅记录上的（一对一的），这样当读者批量借书的时候，系统其实是一本一本分开操作的。具体可见（../logs/conference/storyboard(Extra).txt）里对于借阅的进一步叙述

---

## 遗漏

Q1. 原始需求中没有提到是否需要续借功能的问题

A1. 经过询问协商，确定了对于续借功能的需求。在第二次邮件问询的时候，前台人员向我们叙述了续借的规则(见 ../records/contact/ )

Q2.原始需求没有提到系统可用部署环境的问题

A2. 我们在面谈中提出了这个问题，相关人员告诉我们环境是理想情况，可以不加考虑。

Q3.对于数据及对应系统规则的来源没有涉及

A3.经询问，相关人员表示数据会提供，然而数据库是我方负责，系统规则会提供，同时应当预留接口，便于修改。

Q4.书籍的损毁的确认以及报告，赔偿规则没有提及

A4.经询问，我们知道了赔偿的规则（见 ../records/contact/）；而对于这个损毁报告，我们认为将之加到还书的末尾最为合适，当还到前台的时候，前台会检查并更新响应的损毁度信息和该读者的违约信息（假定图书损毁之后，正常读者会主动选择至前台还书）；否则，则在上架前，前台一样可以检查并通过相应的借阅信息，寻找到最近借阅的“肇事者”。

Q5.各类数据的字段没有明确

A5.经过向维护人员的询问，我们有了大致的了解。基于这些信息，我们后期进行了梳理和抽象。最后的结果可见 类图。