问题说明

**一、需求获取：**

在获取客户需求时，读者和图书馆方代表的需求有较大冲突。读者希望尽可能地保护个人隐私，希望自己的身份证号对于图书馆人员不可见；而图书馆方则需要读者的身份证号作为登录、挂失借书证的唯一凭证，为此须要数据库中读者身份证号的阅览权限。最终，在说明了必要性后，读者代表表示妥协。另一方面，读者希望多数操作都能足不出户地在网上完成，而前台员工代表则坚持续借、赔偿等操作必须在前台或终端完成。在权衡之后，我们接受了图书馆方的方案。

**二、建立数据模型：**

有关借阅历史的存储方式。

为了能够便捷地进行各类查询，新建了借阅历史类，并将其作为读者和书本双方的域进行存储。

**三、有关用户与各角色的继承关系：**

商讨后，决定无论是图书馆内部工作人员还是读者都共享同一个登录入口，因此基于面向对象的思想，将共通的用户名和密码作为父类User类的域储存。

**四、有关财务人员的职责：**

虽然书籍的赔偿额度由后台设定，而特殊书籍的标注由前台在录入时完成，但由于只有财务人员具有这方面的知识，为了能够便捷地设定特殊书籍的赔偿额度，给予财务角色访问并修改相关参数的权限。

**五、建立用况模型和活动图模型：**

1、在不同实体上进行的相同操作：如借书、还书、续借等都建立了用例模型，虽然在终端、前台上借书的逻辑操作基本一致，但由于操作实体的不同，仍然应该采用多个用例。

2、后台人员、前台人员、读者是否需要登录：当读者需要在网站上对自己的账户进行操作时，自然是要登录的。后台人员和财务人员需要获取一定的权限，因此从安全起见，对所有后台人员和财务人员的操作也需要登陆后才能进行。而前台人员默认其处在系统状态内，所以无需登录就可以操作。（但是后来觉得应该将每一类人员的登录作为一个用例实现会更好）

3、系统外部实体的分类：分为了信息系统和支付系统两大类，扫描系统是信息系统的一部分。

4、前后操作的统一：在一些对应的操作中，如借还书，要注意不同外部实体之间交互的统一性，如：判断是否需要交违约金/预付金，要按照判断->缴费->确认缴费->更新系统状态的步骤统一实现。

**六、建立数据流模型：**

1、读者实体是否需要：因为读者所有的操作都是经过终端、前台以及网站这三个外部实体实体，所以并未设置读者的外部实体。

2、进入系统时，是统一先登录再选择操作还是选择操作时由系统提示登录：选择不同的操作后提示登录；（查询借阅信息时需要登录，而查询书籍信息时不要登录）

3、数据库需不需要分块：在零、一层数据流图中数据库未分块，在二、三层数据流图中分块（书籍，读者信息，工作人员信息，系统参数，借阅信息）；

4、挂失与补办的关系（挂失后选择是否要补办）：挂失后可以补办，但不一定选择；

5、续借功能的实现位置：续借功能在前台、终端借书时实现；

6、新书由谁录入，书籍报废的决策：新书最终选择由前台录入，书籍报废由维护人员决定；

7、数据流图不能画的太繁琐：一开始每层图画的较为细致，有很多泡泡，后来将每个图中泡泡数量精简，减少为3~8个。

**七、建立行为模型：**

1、由于无法确定实际操作具体流程，所以状态图中引发状态变化的操作很模糊。毕竟状态图不是最终实现，为了方便理解，初步将其设定为点击按钮

2、实际操作状态太多，无法确定细分程度为了将状态机图最大限度地简明化，只列出了重要的操作

3、很多状态可以同时存在，状态之间的界限模糊，假设这些是不同状态并列出。