## 简述面向对象的软件开发的三个模型的概念和内容

面向对象的软件开发通常涉及三个主要模型，它们分别是：**对象模型、动态模型**和**功能模型**。这些模型共同构成了系统的完整设计。

**(1) 对象模型（Object Model）**

* **概念**：描述系统中的数据结构和静态关系，主要用于表示系统中对象的属性和对象之间的关系。
* **内容**：
  + **类和对象**：定义系统中的实体和它们的属性。
  + **继承**：描述类之间的继承关系。
  + **关联**：描述对象之间的交互关系（如一对一、一对多）。
  + **聚合与组合**：表示对象的整体和部分关系。

**(2) 动态模型（Dynamic Model）**

* **概念**：描述系统的动态行为，尤其是系统中的对象如何随着时间发生状态变化和相互交互。
* **内容**：
  + **状态图（State Diagram）**：显示对象的状态及状态转换。
  + **事件**：触发对象状态转换的外部或内部信号。
  + **生命周期**：描述对象从创建到销毁的整个过程。

**(3) 功能模型（Functional Model）**

* **概念**：描述系统的功能和数据流，主要用于捕获系统的操作逻辑。
* **内容**：
  + **数据流图（DFD）**：显示数据如何在系统中流动以及如何被处理。
  + **操作**：定义输入如何通过操作生成输出。
  + **约束**：定义功能的输入条件和输出条件。

这三个模型分别侧重于系统的静态结构、动态行为和功能逻辑，共同支持面向对象开发的需求分析和设计过程。

## 2、什么是需求获取，什么是需求分析

**(1) 需求获取（Requirement Elicitation）**

* **定义**：是软件开发初期，通过与客户、用户和其他利益相关者的交互，收集和定义系统需求的过程。
* **内容**：
  + **目标**：明确客户和用户的实际需求。
  + **方法**：访谈、问卷调查、头脑风暴、观察业务流程、分析现有系统等。
  + **输出**：初步的需求文档，包含功能需求、性能需求和约束条件。

**(2) 需求分析（Requirement Analysis）**

* **定义**：是在需求获取的基础上，进一步对需求进行分类、验证和建模，以确保需求的完整性、正确性和可行性的过程。
* **内容**：
  + **目标**：澄清模糊需求，发现潜在矛盾，并将需求转化为详细的技术描述。
  + **方法**：
    - 分类：将需求分为功能需求、非功能需求和约束条件。
    - 验证：检查需求是否一致、无冲突、可实现。
    - 建模：使用用例图、数据流图、ER图等工具进行表达。
  + **输出**：需求规格说明书（SRS），为设计和开发提供基础。

需求获取是与客户沟通的阶段，而需求分析则是对这些需求进行加工和规范化的阶段，两者共同构成了需求工程的重要部分。

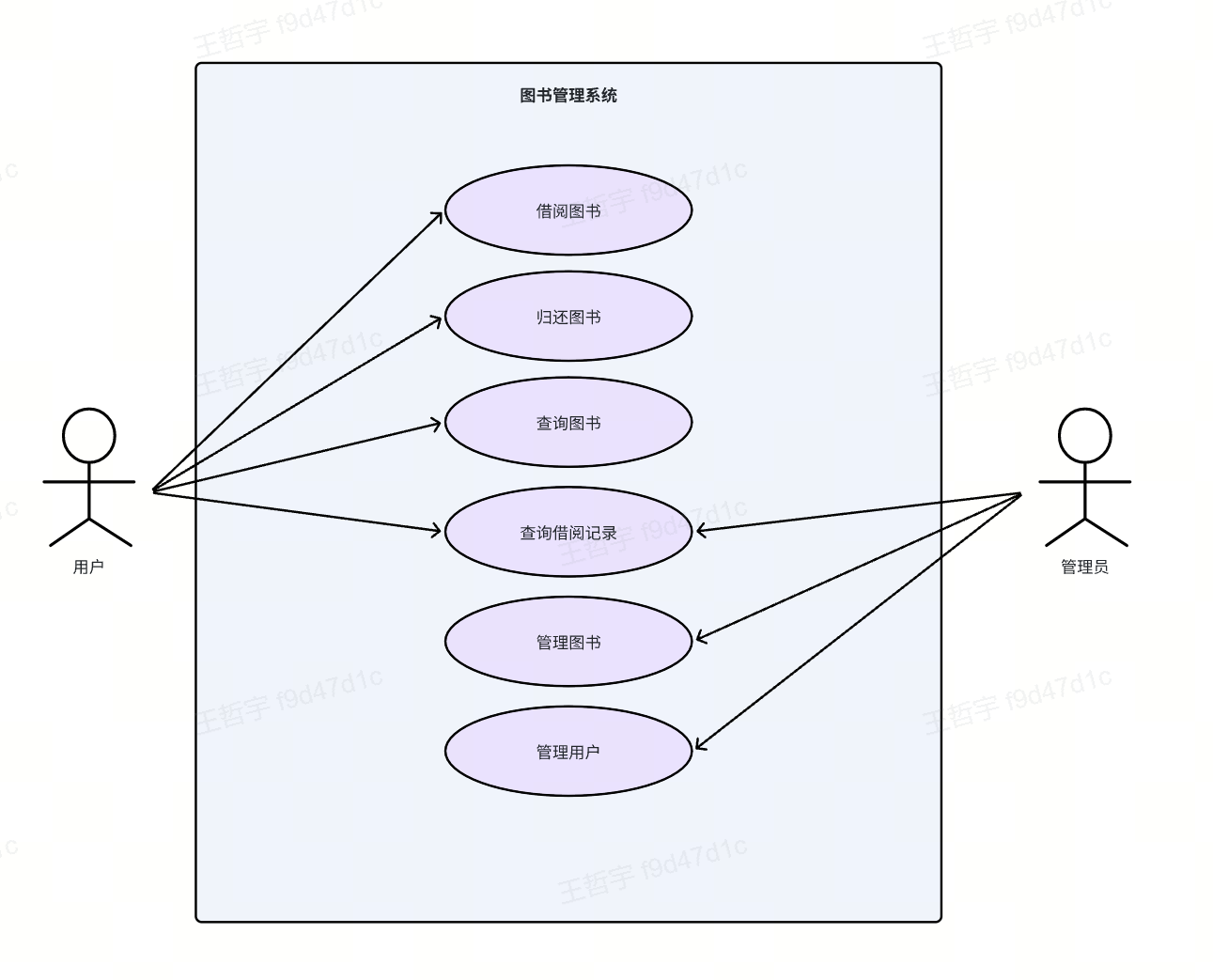
## 3、任选一个或想象一个不太复杂的系统，诸如学籍管理系统、图书管理系统等等诸如此类，用文字简单描述其功能需求，可以用表格列表说明，画出用例图，并选择2个用例画出活动图，并用泳道图组织该活动图

系统选择：图书管理系统

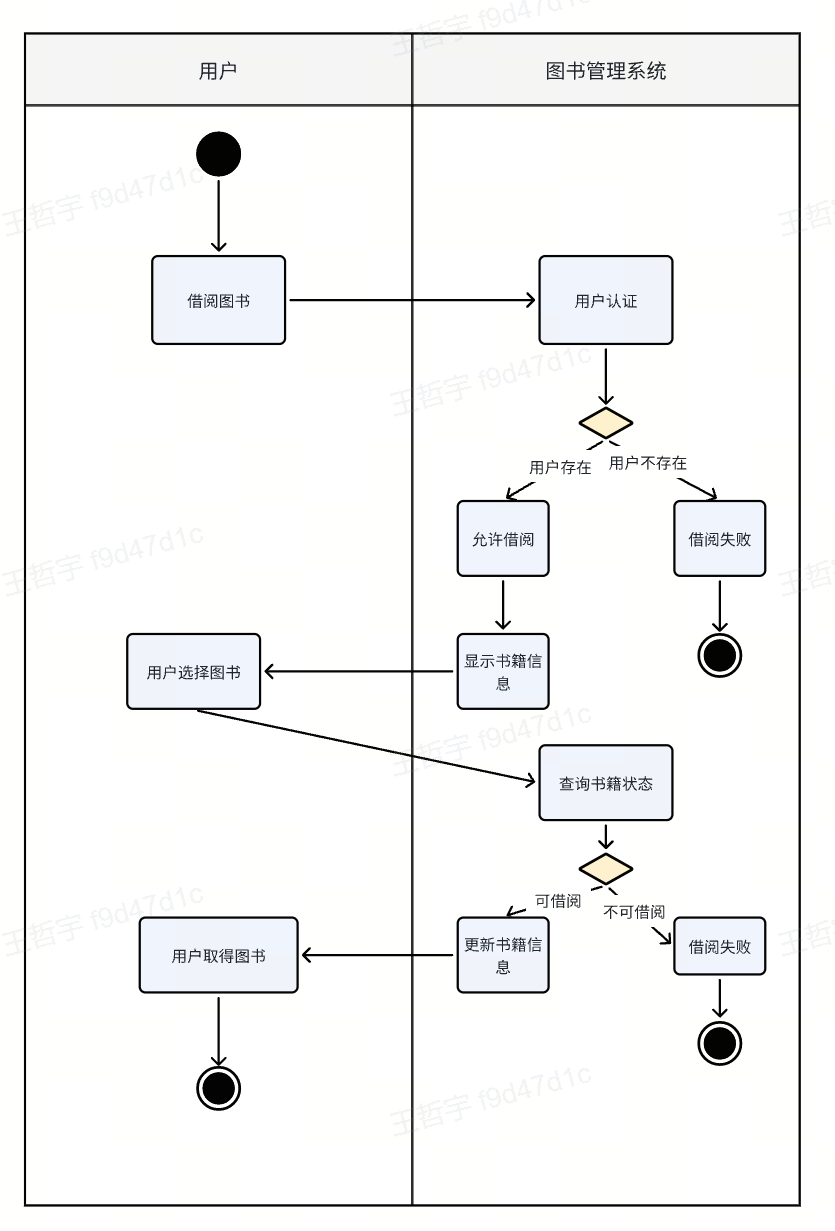
图书管理系统的功能需求如下：

|  |  |
| --- | --- |
| **功能模块** | **功能描述** |
| 图书管理 | 管理图书信息，包括添加、删除、修改和查询图书。 |
| 用户管理 | 管理用户信息，包括注册、删除、查询用户借阅记录等。 |
| 借阅功能 | 用户通过系统借阅图书，管理员负责更新借阅信息。 |
| 归还功能 | 用户归还图书，系统更新归还状态，管理员检查损坏情况。 |
| 图书查询 | 用户可根据书名、作者或分类查询图书信息。 |
| 借阅记录管理 | 记录用户的借阅历史，支持管理员或用户查询。 |

用例图：



用例1：借阅图书



用例2：管理员查看借阅记录

