**详细设计说明书**

# 1引言

## 1.1编写目的

此说明书对《图书馆管理系统》软件系统详细设计实现做了全面细致的阐述，明确该系统整体结构以及相应基本逻辑，帮助项目管理人员和软件开发人员能清楚地了解设计过程，规范实际的开发流程与操作管理。

本说明书的预期读者为开发人员、测试人员、文档编写者、项目管理人员。

本说明书仅适用于图书馆管理系统，任何使用此文档的人员，必须严格按照文档规定及说明及相关法律条款规定，不能擅自修改及非法使用。本开发团队拥有对此文档的最终解释权。

## 1.2背景

说明：

详细设计包括了整个系统各式各样功能实现的细节及其说明。这为后续的开发提供了良好的遵循蓝本，对提高开发效率，减少错误的产生，减轻后期维护和调试的工作量具有十分重大的意义。

1. 待开发的数据库名称：LibraryManagementDatabase

使用此数据库的软件系统名称：图书馆管理系统

1. 软件系统开发项目的任务提出者：西南大学商贸学院

用户：西南大学全体学校学生、管理人员

安装该软件和这个数据库的计算站（中心）：西南大学商贸学院机房

## 1.3定义

登录/注册：用户端进行的身份准入操作，通过即进入系统，失败则重试或退出

借阅/归还：书籍的操作，线上完成数据的变更后，管理员需及时线下一致化。

预约/出让：图书馆资源的使用，具体同上一条。

智能推送：向用户推荐喜爱度高的书籍。

公告管理：管理员发布图书馆公告通知。

奖惩计分：将图书馆规则条例纳入系统。

UserTable：管理用户表

BookGRTable：管理书籍表

ResourceTable：资源管理表

BookTable：书籍信息表

ResourceGRTable：资源信息表

InformTable：公告管理表

## 1.4参考资料

有关的参考文件，如下：

1. 软件开发计划书V1.0；
2. 《图书馆管理系统-需求规格说明书》；《图书馆管理系统-概要设计说明书》等
3. 【1】 GB-T 8567-2006, 《计算机软件文档编制规范》[S].

【2】张海藩，牟永敏《软件工程导论》[M]，北京：清华大学出版社，2013.8

【3】王少锋，《UML面向物件教程》[M]，北京：清华大学出版社，2004.2

【4】王珊，萨师煊，《数据库系统概论》[M]，北京：高等教育出版社，2006.5

# 2程序系统的结构

程序系统的结构介绍，将整个系统序列化、清晰化：

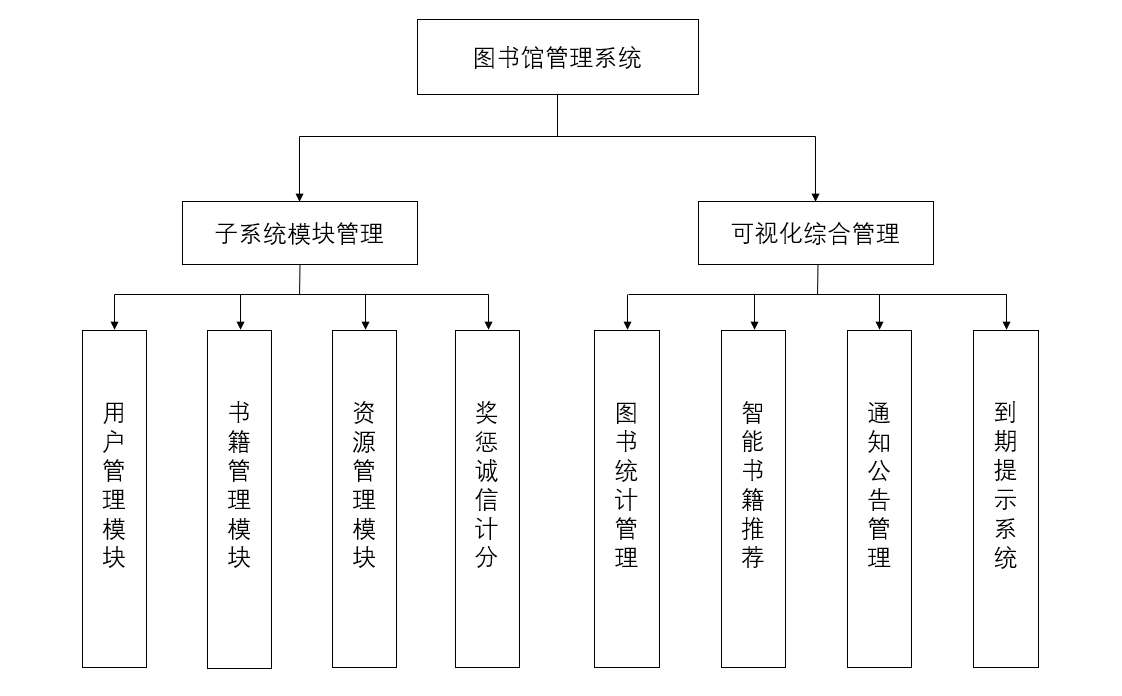


图2-1 系统模块层次结构

表2-1 系统模块与子程序的信息及结构

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 图书馆管理系统 | | | | | | | |
| 模块 | 子系统模块管理 | | | | 可视化综合管理 | | | |
| 系统元素 | 用户管理模块 | 书籍管理模块 | 资源管理模块 | 奖惩诚信计分 | 图书统计管理 | 智能书籍推荐 | 通知公告管理 | 到期提示系统 |
| 标识符 | UserManage | BookManage | ResourceManage | CreditManage | StatisticsManage | RecommandSys | InformManage | DateSys |
| 功能 | 登录/注册 | 借阅/归还 | 预订/出让 | 奖励/惩罚 | 统计各类数据 | 推荐高借阅书籍 | 发布通知 | 日期系统控制 |
| 依赖 | 数据库 | 用户、数据库 | 用户、数据库 | 用户、书籍、资源、数据库 | 数据库 | 数据库 | 数据库 | 时间系统 |

# 3程序1（标识符）设计说明

用户管理模块的设计主要是一个身份的一个审核。只有符合要求的注册账号才能进行登录，使用该系统。同时，为了用户的使用方便，界面友好性是开发时考虑的一个重要问题。在处理这部分功能时，开发团队参考了市面上常用的登录注册板块，结合自身思考，设计了一个良好的登录注册功能模块。‘

书籍管理模块的设计是图书馆管理系统的一个核心应用功能。这部分需要注意的是：不仅是线上的操作，还需要线下同步处理。如，借阅图书和归还图书。一旦线上数据处理完毕，线下在规定的时间内必须进行同步操作，以保持整个系统的一致性。

资源管理模块，同上一个模块的设计思想一样，关键还是在于线上线下同步的问题。

奖惩诚信计分模块，这部分的设计参考了大部分的系统的一个积分模块。考虑到在实际的使用场景中，需要对一些不遵守图书馆规章制度的人员进行处罚，以维护管理秩序。因此，本模块很好的实现了这一现实需求。此外，整个模块的实现也较为简单，不会使整个系统的逻辑和结构变得很复杂。

图书统计管理模块，主要的设计思路是利用数据库中相应的表单，抽离出相关数据，进行可视化统计，即从海量数据中挖掘出有价值的数据，帮助图书管理员进行服务上的改进。

智能书籍推荐模块，这部分和上一个模块的内核一样，同样是基于数据库相关数据进行处理，然后形成逻辑操作的过程。不同的是，这部分的操作更加的抽象和更具交互性。此外，这部分使用的相关技术也较为密集，设计时需将其内部结构合理设计。

通知公告管理模块和到期提示系统均是将数据发布到平台上，不管公告的主动发送，还是到期信息的被动提醒，都是利用了数据库的基础功能，这里的设计较为简略。

## 3.1程序描述

程序描述了该程序的基本信息以及运行时的状态，能够帮助开发者和维护者后期进行调试和维护。

表3-1 程序模块描述

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 程序模块 | 目的意义 | 特点 | | | | |
| 常驻内存 | 子程序 | 可重用 | 覆盖要求 | 处理类型 |
| 用户管理模块 | 用户准入和维护 |  |  |  | √ | 并发 |
| 书籍管理模块 | 书籍管理 |  |  | √ |  | 并发 |
| 资源管理模块 | 资源管理 |  |  | √ |  | 并发 |
| 奖惩诚信计分 | 落实管理制度 | √ | √ |  | √ | 并发 |
| 图书统计管理 | 提高服务质量 | √ |  |  |  | 顺序 |
| 智能书籍推荐 | 提高服务质量 | √ | √ |  |  | 顺序 |
| 通知公告管理 | 提高服务质量 |  |  | √ | √ | 顺序 |
| 到期提示系统 | 提高服务质量 |  |  | √ | √ | 顺序 |

## 3.2功能

说明该程序应具有的功能，采用IPO图（即输入一处理一输出图）的形式形象展示如下：

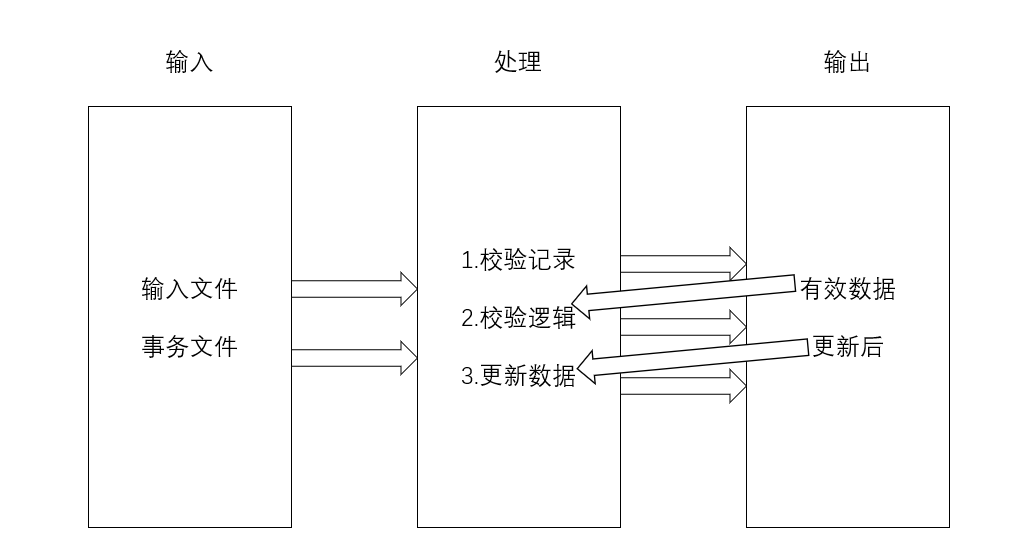


图3-1 程序宏观IPO

|  |  |
| --- | --- |
| 系统：图书馆管理系统 | 作者：余帅文 |
| 模块：基本功能 | 日期：2021/5/1 |
| 编号：001 |  |
| 被调用：main() | 调用：databaseCall() |
| 输入：text | 输出：message |
| 处理：逻辑判断，返回 |  |
| 局部数据元素：datastruct | 注释：单元注释 |

图3-2 改进的程序IPO图—基本功能

|  |  |
| --- | --- |
| 系统：图书馆管理系统 | 作者：余帅文 |
| 模块：核心功能 | 日期：2021/5/3 |
| 编号：001 |  |
| 被调用：main() | 调用：managementCall() |
| 输入：valuedata | 输出：message |
| 处理：逻辑判断，输出 |  |
| 局部数据元素：datastruct | 注释：单元注释 |

图3-3 改进的程序IPO图—核心功能

## 3.3性能

该系统服务于一个大学的全体师生，用户规模庞大，因此所需要的性能要求也比较严格，以下列举其中具有代表性的几个方面的性能要求：

表3-2 性能说明

|  |  |  |
| --- | --- | --- |
| 图 | 性能要求 | 详细描述 |
| 书 | 高并发 | 庞大的用户群体同时访问服务器，需要服务器拥有高并发处理的能力 |
| 馆 | 低延迟 | 每一个功能操作为了满足用户的体验，需要将延迟时间控制在0.1s以内 |
| 管 | 冗余保障 | 数据库运行要确保稳定，若出现不可预测的错误，需立即启动冗余备份 |
| 理 | 保密安全 | 大量的用户信息需要得到安全性的保障 |

## 3.4输人项

输入项的基本信息从侧面反映了一个系统的可行性和兼容性，是评价一个系统好坏的标准之一。

表3-3 输入项信息

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 特性 | | | | | | | | | |
| 输入项 | 名称 | 标识 | 数据类型格式 | 数据有效范围 | 方式 | 数量 | 频度 | 输入媒体 | 来源 | 安全保密条件 |
| 1 | 登录 | username/password | varchar | 40 | 键入 | 用户数 | 10% | 键盘 | 前端 | 高 |
| 2 | 操作 | operation | varchar | 8 | 点击 | 操作数 | 30%~40% | 触控/鼠标 | 前端 | 低 |
| 3 | 跳转 | linked | varchar | 4 | 点击 | 点击量 | 20% | 触控/鼠标 | 前端 | 低 |
| 4 | 统计 | statistics | int | 0~1e6 | 读 | 记录数 | 20%~30% | 磁盘 | 数据库/后端 | 中 |

## 3.5输出项

基于输入项得到的输出项便是对系统的一个客观的描述。

表3-4 输出项信息

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 特性 | | | | | | | | | |
| 输入项 | 名称 | 标识 | 数据类型格式 | 数据有效范围 | 形式 | 数量 | 频度 | 输出媒体 | 说明 | 安全保密条件 |
| 1 | 登录 | username/password | varchar | 40 | 跳转 | 用户数 | 10% | 显示器 | 进入系统 | 高 |
| 2 | 操作 | operation | varchar | 8 | 响应 | 操作数 | 30%~40% | 显示器 | 完成操作 | 低 |
| 3 | 跳转 | linked | varchar | 4 | 页面 | 点击量 | 20% | 显示器 | 转换页面 | 低 |
| 4 | 统计 | statistics | varchar | 255 | 图 | 统计量 | 20%~30% | 显示器 | 可视化 | 中 |

## 3.6算法

本系统所用算法主要应用在保密、并发以及推荐系统三个方面。

①数据加密保护算法：MD5码

**1数据填充**

对消息进行数据填充，使消息的长度对512取模得448，设消息长度为X，即满足X mod 512=448。根据此公式得出需要填充的数据长度。

填充方法：在消息后面进行填充，填充第一位为1，其余为0。

**2添加消息长度**

在第一步结果之后再填充上原消息的长度，可用来进行的存储长度为64位。如果消息长度大于264，则只使用其低64位的值，即（消息长度 对 264取模）。

在此步骤进行完毕后，最终消息长度就是512的整数倍。

**3数据处理**

准备需要用到的数据：

* 4个常数： A = 0x67452301, B = 0x0EFCDAB89, C = 0x98BADCFE, D = 0x10325476;
* 4个函数：F(X,Y,Z)=(X & Y) | ((~X) & Z); G(X,Y,Z)=(X & Z) | (Y & (~Z));  H(X,Y,Z)=X ^ Y ^ Z; I(X,Y,Z)=Y ^ (X | (~Z));

把消息分以512位为一分组进行处理，每一个分组进行4轮变换，以上面所说4个常数为起始变量进行计算，重新输出4个变量，以这4个变量再进行下一分组的运算，如果已经是最后一个分组，则这4个变量为最后的结果，即MD5值。

②并发性算法

**1并发的概念：**

　　指网站在同一时间访问的人数，人数越大，瞬间带宽要求更高。

**2服务器并发量分为：**

　　1.业务并发用户数；2.最大并发访问数；3.系统用户数；4.同时在线用户数；

**3估算业务并发量的公式：**

　　   
　　其中：C是平均的业务并发用户数、n是login*session的数量、L是login* session的平均长度、T是指考察的时间段长度、C^是指业务并发用户数的峰值。

③推荐算法：协同过滤算法（Collaborative Filtering）

CF算法的原理是汇总所有<user,item>的行为对, 利用集体智慧做推荐。其原理很像朋友推荐, 比如通过对用户喜欢的item进行分析, 发现用户A和用户B很像(他们都喜欢差不多的东西), 用户B喜欢了某个item, 而用户A没有喜欢, 那么就把这个item推荐给用户A。(User-Based CF)

当然, 还有另外一个维度的协同推荐。即对比所有数据, 发现itemA和itemB很像(他们被差不多的人喜欢), 那么就把用户A喜欢的所有item, 将这些item类似的item列表拉出来, 作为被推荐候选推荐给用户A。(Item-Based CF)

该公式要计算用户i和用户j之间的相似度, I(ij)是代表用户i和用户j共同评价过的物品, R(i,x)代表用户i对物品x的评分, R(i)头上有一杠的代表用户i所有评分的平均分, 之所以要减去平均分是因为有的用户打分严有的松, 归一化用户打分避免相互影响。

本系统将推荐算法整合用于图书的喜爱度推荐，能够较好的提高服务质量，获得良好反馈效果。

## 3.7流程逻辑

本系统的流程逻辑十分自然。首先是用户进行一个登录/注册，获得使用权限。当然，普通用户和管理员用户所获得的权限是不一样的。接着，普通用户可以进行图书的借阅/归还、资源的预定/出让等操作；而管理员则可以进行数据库管理和公告的发布。最后，线上系统要和线下资源进行同步对接，保持一致性。

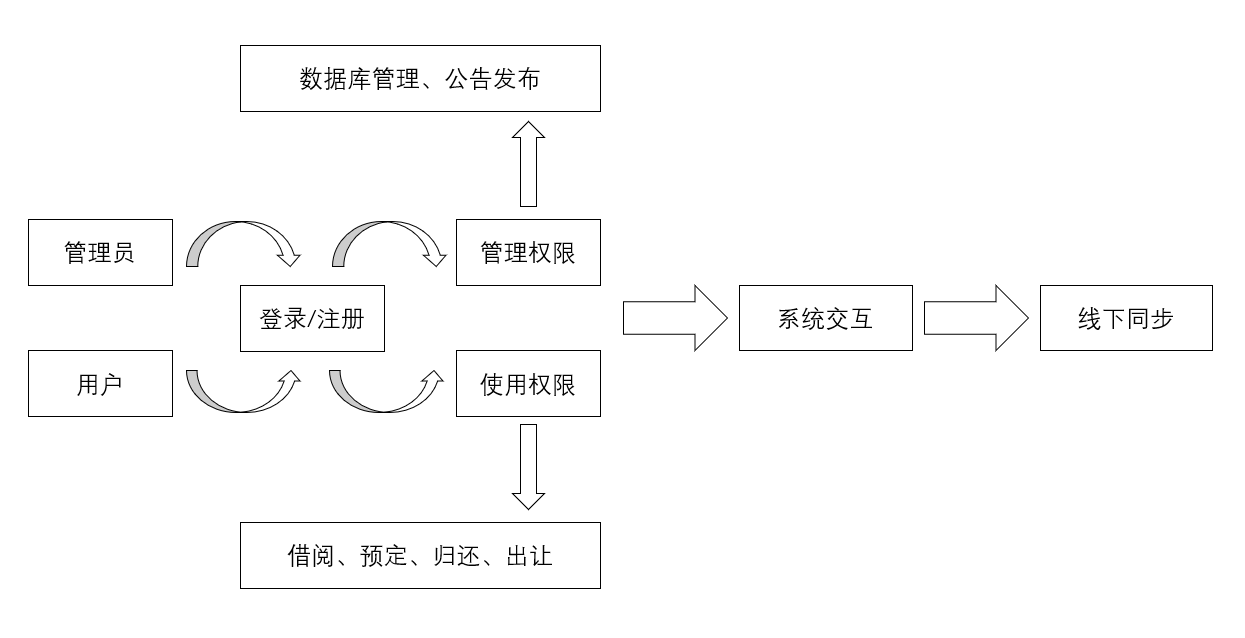


图3-4 逻辑流程图

表3-5 流程判定表

|  |  |  |  |
| --- | --- | --- | --- |
| 程序 | 取值判定 | | |
| 登录/注册 | 管理员 | 普通用户 | 普通用户 |
| 公告/数据库管理 | 有权限 | 无权限 | 无权限 |
| 资源操作 | —— | 使用 | 归还 |
| 是否符合规定 | —— | 有余量/计分好 | 没逾期 |
| 结果 | √ | √ | √ |

## 3.8接口

本系统的接口包括内部接口和外部接口，具体信息如下：

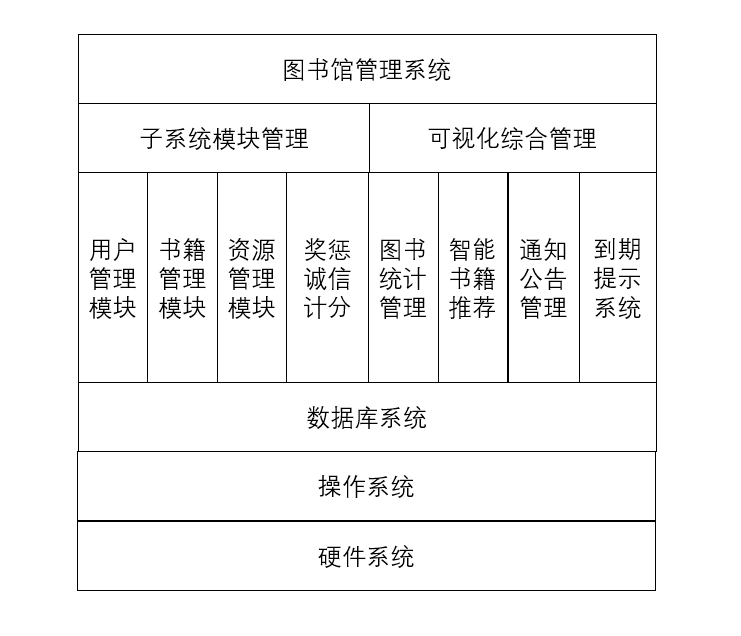


图3-5 模块结构隶属关系

表3-6 参数赋值与调用方式

|  |  |  |
| --- | --- | --- |
|  | 参数赋值 | 调用方式 |
| 用户管理模块 | 账号密码 | 被动监听 |
| 书籍管理模块 | 书籍id号 | 主动调用 |
| 资源管理模块 | 资源id号 | 主动调用 |
| 奖惩诚信计分 | 用户id号 | 被动监听 |
| 图书统计管理 | 书籍表count | 自动运行 |
| 智能书籍推荐 | abstract data | 自动运行 |
| 通知公告管理 | text | 主动调用 |
| 到期提示系统 | date | 被动监听 |

表3-7 程序关联数据结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | User | BookGR | Resource | Book | ResourceGR | Inform |
| 用户管理 | √ |  |  |  |  |  |
| 书籍管理 | √ | √ |  | √ |  |  |
| 资源管理 |  |  | √ |  | √ |  |
| 奖惩诚信 | √ |  |  |  |  |  |
| 图书统计 |  | √ |  | √ |  |  |
| 智能书籍 | √ | √ |  | √ |  |  |
| 通知公告 |  |  |  |  |  | √ |
| 到期提示 | √ | √ | √ | √ | √ |  |

## 3.9存储分配

本系统的功能很大程度上依赖于存储的效能，因此如何设计存储分配是一个至关重要的问题。团队采用了“云数据库+机房+临时存储（端系统）”的一体化做法，更好地提高了存储效能，也保证了一定的安全性。

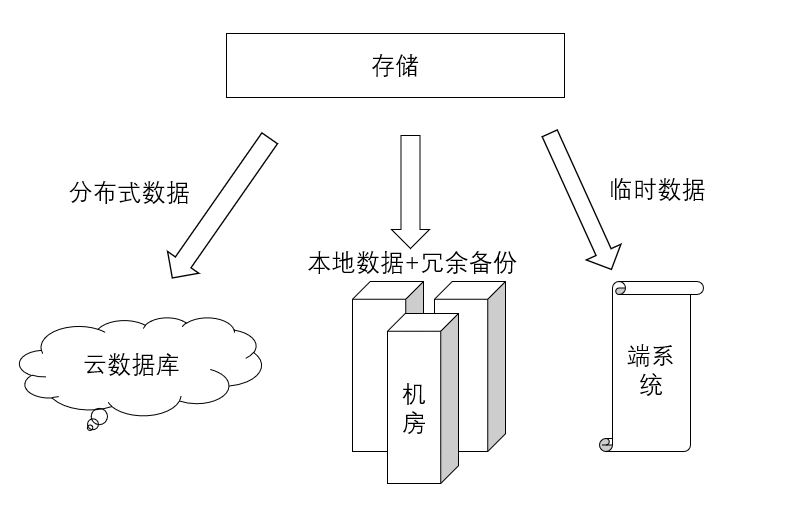


图3-6 存储分配图

## 3.10注释设计

1. 加在模块首部的注释：模块注释，解释接下来模块的功能和基本结构
2. 加在各分枝点处的注释：分支辨别，根据条件厘清逻辑关系
3. 对各变量的功能、范围、缺省条件等所加的注释：功能注释，解释对应功能
4. 对使用的逻辑所加的注释：将逻辑抽象出来，方便理解

## 3.11限制条件

数据库运行情况；网络流通情况；

## 3.12测试计划

该系统可分别进行单元测试、功能测试、系统测试、接口测试，尽量找出其可能存在的错误以及漏洞。从测试技术要求上来看，需要根据对应的功能模块设计相应的测试用例。将这些测试用例逐一输入，将结果与期望对比，找出bug所在并记录。在这个过程中要注意进度安排和人员的调度。当bug被测试人员测试出后，将其打包为任务分配给技术人员进行调试修改。此外，还要关心整个系统的运行环境和设备驱动条件、桩模块的处理等等。

## 3.13尚未解决的问题

无

# 4程序2（标识符）设计说明

程序设计要考虑其功效以及实现方法，两者要匹配，以期达到资源利用最大化的效果。一些可重用的模块最好将其独立出来，降低耦合，提高内聚。下面用一个简洁的矩阵图展示主要部分程序如下：

表4-1 程序设计矩阵图

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 前端数据收集 | 后端逻辑处理 | 数据库访问 | 子模块调用 |
| 登录注册 | √ |  | √ |  |
| 借阅/归还 |  | √ | √ | √ |
| 资源管理 |  | √ | √ | √ |
| 智能推荐 | √ | √ | √ | √ |
| 诚信分机制 |  | √ | √ |  |
| 可视化统计 | √ |  | √ |  |
| 公告发布 | √ |  | √ | √ |