



西北师范大学

软件系统设计说明书

题 目： 背包问题知识社区系统

学 院： 计算机科学与工程学院

专 业： 计算机科学与技术

班 级： 2019 级卓越工程师班

团队名称： 奋起上进组

团队成员： 曹永兴 李斌 尚洁 张蓉星

目录

1 引言	3
1.1 编写目的.....	3
1.2 背景.....	3
1.3 定义.....	4
1.4 参考资料.....	4
2 总体设计.....	5
2.1 需求规定.....	5
2.2 运行环境.....	6
2.3 基本设计概念及处理流程.....	7
2.4 结构.....	8
2.5 管理员处理过程.....	8
2.6 异常处理.....	8
2.7 设计模式.....	9
2.8 安全设计.....	9
2.9 重用方案.....	10
2.10 关键类的重点服务.....	10
3 系统数据结构设计.....	11
3.1 数据库逻辑结构.....	11
3.2 状态图.....	11
3.3 系统数据字典.....	14

1 引言

1.1 编写目的

在系统需求分析的基础上，对整个背包知识讨论系统的功能划分，机器设备配置，数据的逻辑结构设计以及软件系统的总体结构设计进行说明。

1.2 背景

截至 2021 年，我国 GDP 已经仅次于美国，成为当今世界上第二大强国。但是从另一方面我们也要看到之所以能够取得如此骄人的成绩，其原因还是在于教育。然而在互联网时代，随着科技的快速发展，传统意义上的线下教育也逐渐走向了网络，在此过程中，知识分享平台的出现对于提高全民综合素养更是有着重要意义。知识交流可以辅助多人交流、协同进行知识创新的环境或系统，而网络资源层出不穷，出现各种社区交流系统，而各个系统的知识面涉猎较广泛，并未出现专有的背包问题讨论平台，正是由于这一点，对于初学者来说，大而全的平台并不是一个好的选择，而背包问题作为组合优化和计算机理论科学的基本问题之一，在计算机理论和运筹学等领域中扮演者重要角色，具有很必要的讨论价值，设计一个小而全的背包问题讨论系统显得尤为必要。

另外手机 APP 的多方优势：

1、大的流量入口

随着智能手机的普及，中国已经成为移动互联网规模大国，全球移动互联网市场。手机 APP 就是大流量入口，抢占用户的手机桌面已经成为当前营销的重点。

2、更加及时便利

对比传统的网站、论坛，手机 APP 的优势是无可比拟的，可以随时随地的打开手机 APP 浏览使用，非常方便。

3、APP 的宣传效果更佳

通过手机 APP，可以将产品以精美的图文、视频等综合形式进行展示，提升产品的品牌力，提高用户对产品的信任力。

4、提高用户粘性

可以通过应用公园在 APP 内设置社区论坛版块，用户可以自由发帖交流，也可以互相添加好友，构建行业人脉圈。同时通过资讯系统，不定期推送分享新的行业动态、资讯等。

1.3 定义

(1) 客户

与我们签订合同，要求购买产品的人。软件的开发以他们的要求为基础。

(2) 用户

- 在校大学生或高校老师
- 需要进行代码编写的程序员

1.4 参考资料

[1] 软件工程导论(第五版)，张海潘，清华大学出版社

[2] 数据库原理及应用

2 总体设计

2.1 需求规定

表 2-1 产品功能表

功能	概述	功能性分类
用户注册	进入 app 后,用户可以通过自己的手机号进行注册。	注册
算法导航	应用界面提供典型算法、开源数据集和文献资源。	浏览
发现优秀文章	主要包括用户查看优秀文章	浏览
资源上传及回答 问题	注册用户可以上传自己的资源和回答其他用户的问题	上传
消息	可查看并回复其他人发来的消息	交流
用户个人信息	有用户个人的基本信息	浏览与修改

(1) 主要输入输出要求

①输入项目：

- 1)所查询的算法(算法类别等)
- 2)资源信息(开源数据集，典型算法源码等)
- 3)文章信息（文章具体内容，编写人等）
- 4)用户信息

②输出项目：

- 1) 算法与资源信息
- 2) 文章内容
- 3) 用户信息

功能描述：

- 一、 用户注册 :用户注册后使用。
- 二、 一级界面导航 :按照主要功能分为首页、发现优秀文章界面、资源上传及回答问题界面、消息界面和用户个人信息界面共 5 个总界面导航。
- 三、 二级首页界面导航: 可在首页分别获取典型算法、开源数据集和文献资源。其中典型算法按照算法类型设置了三级导航，方便用户查看某一类型的算法求解。
- 四、 文章编写功能: 用户可编写文章分享自己解决背包问题的算法，分享自己的思路和心得等。注：暂不发布的文章可存为草稿。
- 五、 文章详情 :用户能够查看文章的详情，包括文章的标题、发布人、发布时间、以及该文章的点赞、收藏及评论数量等。
- 六、 问题的发布与回答: 用户可将自己的问题发布到平台进行讨论，也可回答其他用户的问题。
- 七、 搜索 :能够更快更方便的在短时间找到自己想要获取的资源或了解的算法。系统能够帮助用户记住自己的历史搜索记录。
- 八、 推荐相关文章:用户在查看某一类型算法后，系统可以向用户推荐其它相关算法求解文章给用户。
- 九、 热门文章显示: 系统会向用户推荐浏览量较多的优秀文章，方便用户学习。
- 十、 资源上传: 用户上传优秀相关算法数据集、获取到的文献资源等。
- 十一、 消息查看及回复: 可与其他学者私发信息进行问题讨论。
- 十二、 用户信息查看: 可查看修改自己的信息，也可查看其他用户的基本信息。
- 十三、 资源审核: 资源上传后需要接受审核，营造良好知识社区环境。
- 十四、 文章收藏: 用户认为较好的文章可进行收藏。
- 十五、 文章查看记录: 在个人信息界面设置了历史记录，方便用户再次查看之前浏览过的文章。

2.2 运行环境

设备:计算机, 手机

操作系统: Windows 操作系统, Android 系统

数据库管理系统软件: MySQL server
接口:局域网与互联网接口, 软件与操作系统接口。

2.3 基本设计概念及处理流程

1.E-R 图

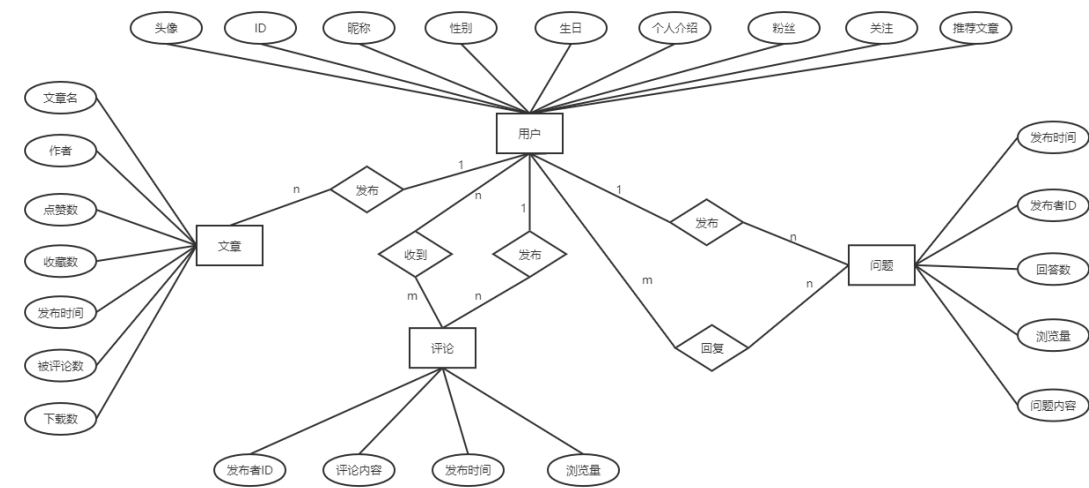


图 1 E-R 图

数据流图如下:

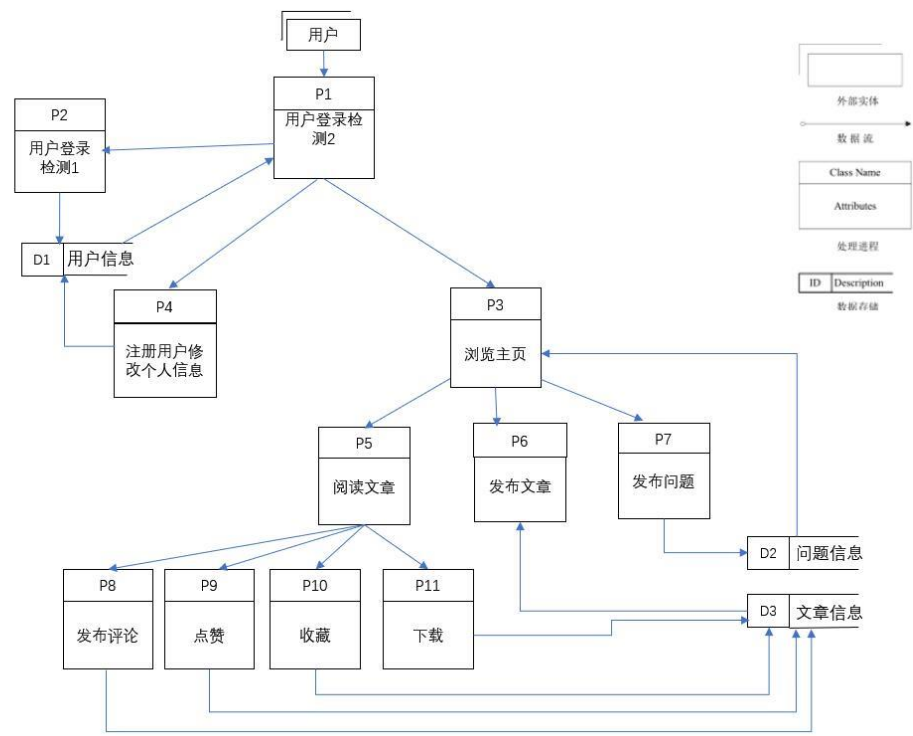


图 2 数据流图

2.4 结构

根据该背包知识讨论系统软件的实际需求，可以将图书借阅系统划分为 6 个部分，各个部分的具体功能的系统功能结构图如图所示：

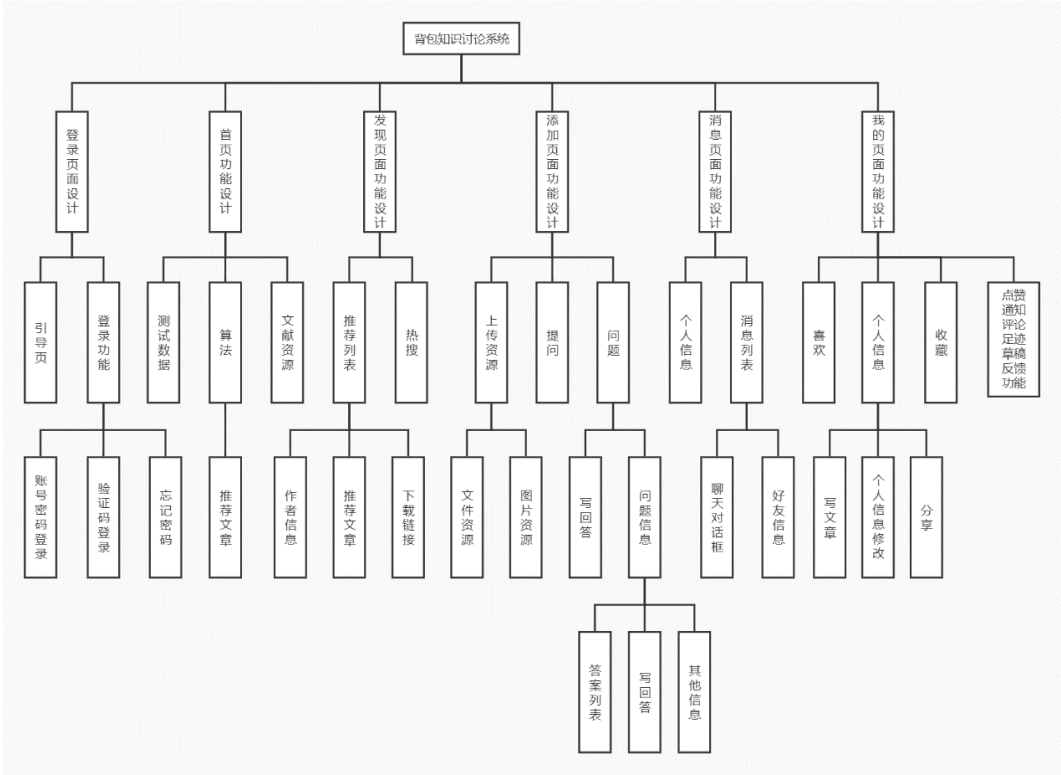


图 3 功能层次图

2.5 管理员处理过程

- (1) 当用户上传资源时，管理员审核资源信息。
- (2) 当有新用户注册时，管理员录入新用户信息。
- (3) 当有老用户须注销系统，管理员将老用户信息删除、接口设计。

2.6 异常处理

异常处理(又称为错误处理)是编程语言或计算机硬件里的一种机制，用于处理软件或信息系统中出现的异常状况(即超出程序正常执行流程的某些特殊条件)。其功能提供了处理程序运行时出现的任何意外或异常情况的方法。本系统的异常处理机制如下：

- (1) 在应用程序遇到异常情况（如被零除情况或内存不足警告）时，就会产生异常。发生异常时，控制流立即跳转到关联的异常处理程序（如果存在）。

(2) 如果给定异常没有异常处理程序，则程序将停止执行，并显示一条错误信息。

(3) 可能导致异常的操作通过 `try` 关键字来执行。

(4) 程序可以使用 `throw` 关键字显式地引发异常。

(5) 异常对象包含有关错误的详细信息，其中包括调用堆栈的状态以及有关错误的文本说明。

(6) 即使引发了异常，`finally` 块中的代码也会执行，从而使程序可以释放资源。

(7) 异常处理使用 `try`、`catch` 和 `finally` 关键字来尝试可能未成功的操作，处理失败，以及在事后清理资源。异常处理通常是防止未知错误产生所采取的处理措施。

异常由公共语言运行库(CLR)、第三方库或使用 `throw` 关键字的应用程序代码生成，也可以用自定义的异常处理类来扩展 `php` 内置的异常处理类。

2.7 设计模式

MVC 设计模式一般指 MVC 框架，M (Model) 指数据模型层，V (View) 指视图层，C (Controller) 指控制层。使用 MVC 的目的是将 M 和 V 的实现代码分离，使同一个程序可以有不同的表现形式。其中，View 的定义比较清晰，就是用户界面。

在 Web 项目的开发中，能够及时、正确地响应用户的请求是非常重要的。用户在网页上单击一个 URL 路径，这对 Web 服务器来说，相当于用户发送了一个请求。而获取请求后如何解析用户的输入，并执行相关处理逻辑，最终跳转至正确的页面显示反馈结果，这些工作往往是控制层 (Controller) 来完成的。

在请求的过程中，用户的信息被封装在 User 实体类中，该实体类在 Web 项目中属于数据模型层 (Model)。在请求显示阶段，跳转的结果网页就属于视图层 (View)。

像这样，控制层负责前台与后台的交互，数据模型层封装用户的输入/输出数据，视图层选择恰当的视图来显示最终的执行结果，这样的层次分明的软件开发和处理流程被称为 MVC 模式。

2.8 安全设计

在背包问题知识社区系统中，参考 OSI 的七层协议，从网络级安全、传输级安全、系统级安全和应用级安全等几方面进行考虑，主要遵循下面的设计原则：

(1) 标识与确认：任何用户访问系统资源，必须得到系统的身份认证以及身份标识，如用户的数据证书、用户号码、密码。当用户信息与确认信息一致时，才能获准访问系统。

(2) 授权：对系统资源，包括程序、数据文件、数据库等，根据其特性定义其保护等级；对不同的用户，规定不同的访问资源权限，系统将根据用户权限，授予其不同等级的系统资源的权限。

(3) 日志：为了保护数据资源的安全，在系统中所保护的资源进行任何存取操作，都做相应的记录，形成日志存档，完成基本的审计功能。

(4) 加密：为了保护数据资源的安全，在系统中所在网络中传输的信息必须经过高强度的加密处理来保证数据的安全性。

2.9 重用方案

软件重用方案：

- 知识重用方案：学习过的知识可以得到充分利用，比如数据库，JAVAAEE，WEB 前端技术等。对于我们软件需求分析以及后期软件开发都有很好的作用。
- 方法和标准的重用方案：采用面向对象的方法，对于标准的重用，我们采用国家规定的软件开发规范。整体遵守代码规范，对每个人的编码都进行规范化，使得后期维护方便快捷，代码整体结构清晰。定义通用的接口以及方法，使用过程中直接调用即可。
- 软件成分的重用方案：在后期编码的过程中，使用调用库函数，实现代码重用，可以大大提高代码的效率。对于编写的源代码、用户界面的设计、数据等都可以重用，比如源代码的编写中，所要用到的一些前端框架，我们就可以结合自身本系统的需求进行分析，然后进行框架整合，这样就减少了我们的编码工作量，有助于加快项目进度。在 UI 界面中，可以重用之前的原型设计中设计好的界面，对应进行设计，因为已经对原型进行了调研和试用，所以重用原型设计来设计界面，可以保证我们最后研发出来的产品是满足用户审美的，是符合用户需求的。对于数据重用，比如数据库中表的设计，我们可以对逻辑分析中的 E-R 图进行加工，根据 E-R 图进行建表等工作，根据逻辑分析设计数据库。
- 类构件实现软件重用方案设计：为了提高重用的效果,关键是设计一个合理的、具有一定深度的类构件继承层次结构,这样每个子类在继承父类的属性和服务的基础上,只需要加入少量新属性和新服务,这不仅降低了每个类构件的接口复杂度,表现出一个清晰的进化过程,提高了每个子类的可理解性,而且为软件开发人员提供了更多可重用的类构件。

1. Android viewpager 之 view 的重用：

用一个链表,把移除出去的 view 保存起来,在初始 view 的时候,判断链表之中是否有待重用的 view,,有就重用,没有就创建;

2. 接口：

直接调用一个 xml 文件，把调用的 xml 当做一个整体。

3. 重用应用资源：

在应用退出时，只 finish 主 Activity，而不杀掉应用进程。这样在应用退出后热启时可以重用应用资源。

2.10 关键类的重点服务

软件重用方案主要是在源代码级，通过类的继承、调用来实现。其实可重用的范围是很大的，如设计的重用，测试用例的重用，可运行的代码的重用等。将

来会扩大重用的粒度，在框架基础上，进一步根据我们项目本身的软硬件环境定制出一个适用于我们系统的框架，将可以极大地提高软件的开发和维护的效率。

在本次设计中，我们的系统主要是用户模块；首先创建一个类，用于用户的注册，随后创建另一个类，作为登录界面。用户界面主要有两个功能，发现优秀文章界面、资源上传及回答问题界面、消息界面和用户个人信息界面。页面均在一个基础的框架增加内容，编写时首先调用框架，进行重用。控制器负责调用各项功能，在进行操作时，均需对其进行重用。

3 系统数据结构设计

3.1 数据库逻辑结构

E-R 图提供了表示实体类型、属性和联系的方法，用来描述现实世界的概念模型，修改 E-R 图，得出数据库逻辑结构。如图 4 所示。

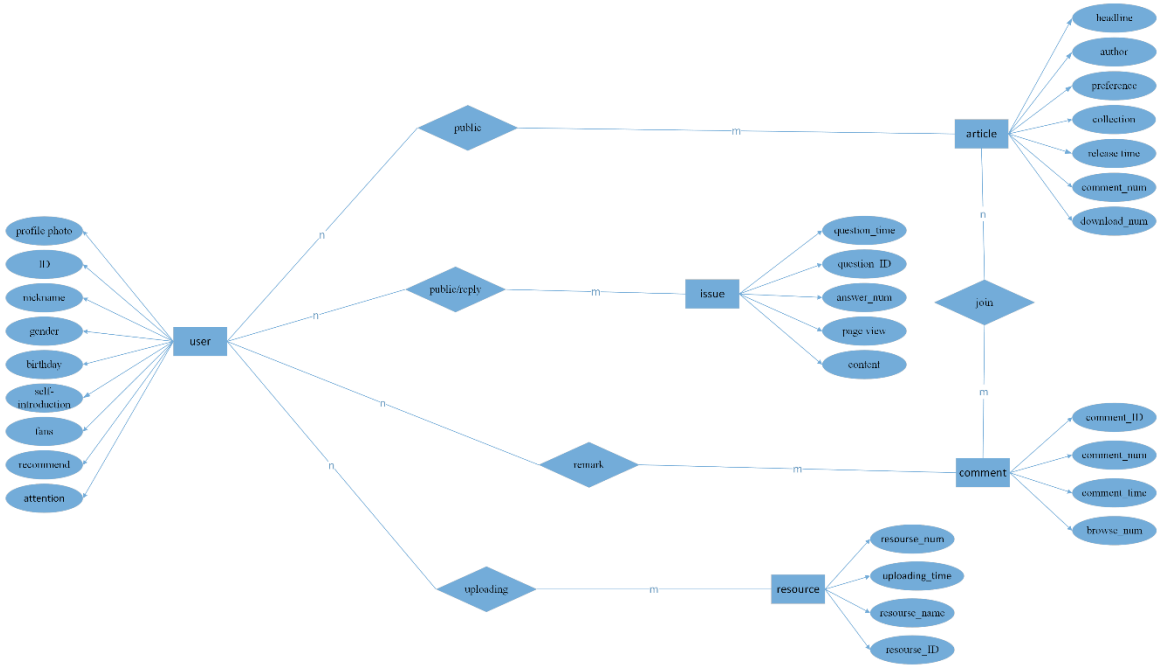


图 4 数据逻辑结构图

3.2 状态图

各页面状态图如下：

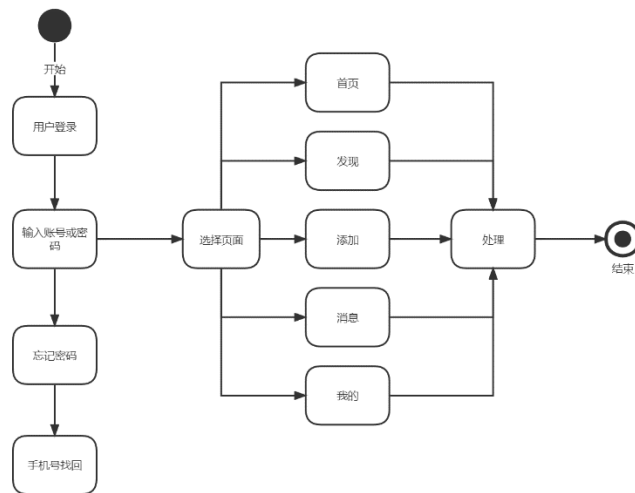


图 5 总体状态图

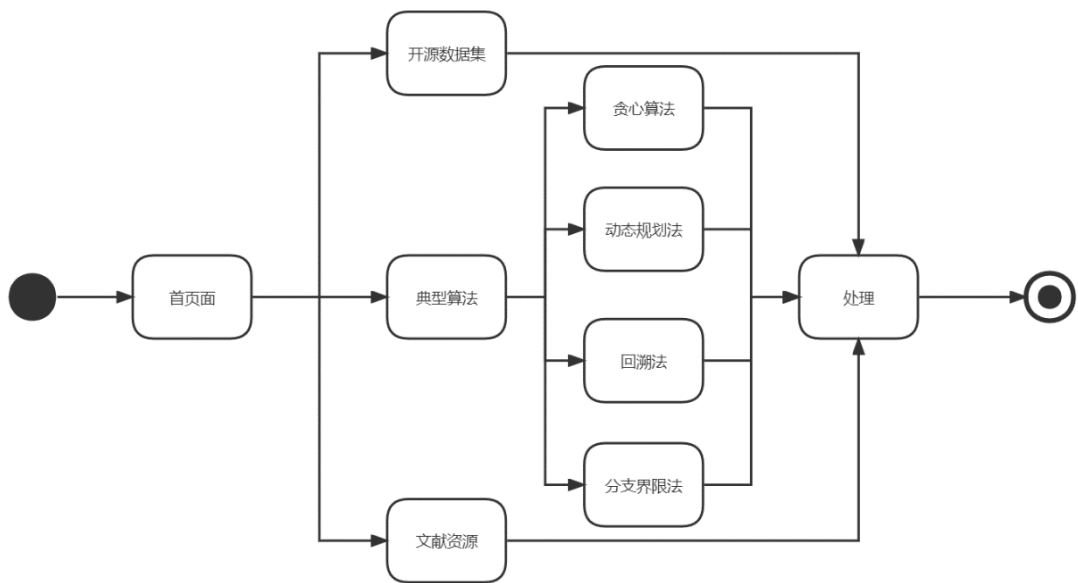


图 6 首页状态图

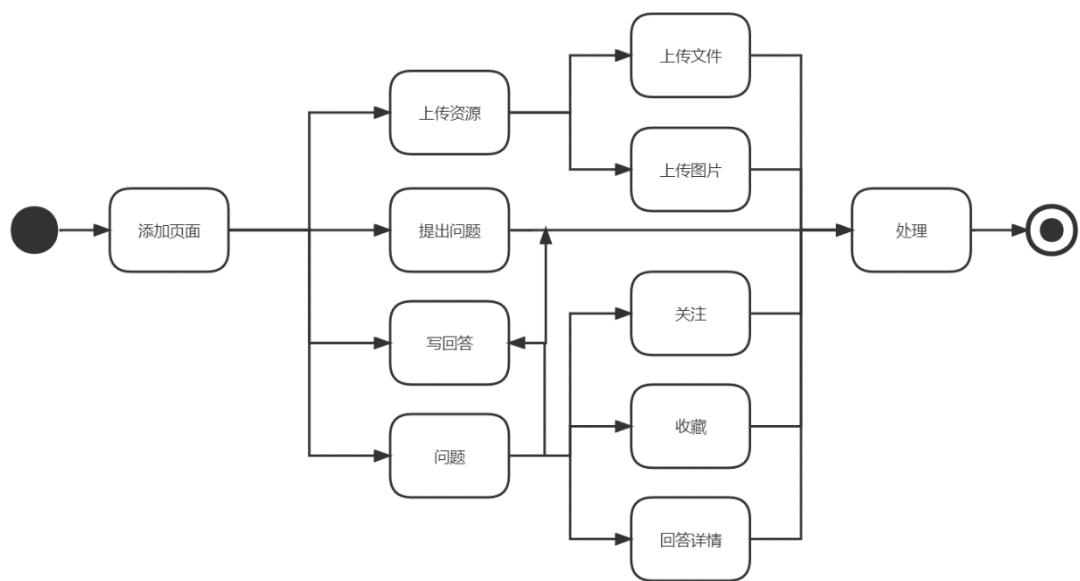


图 7 添加页面状态图

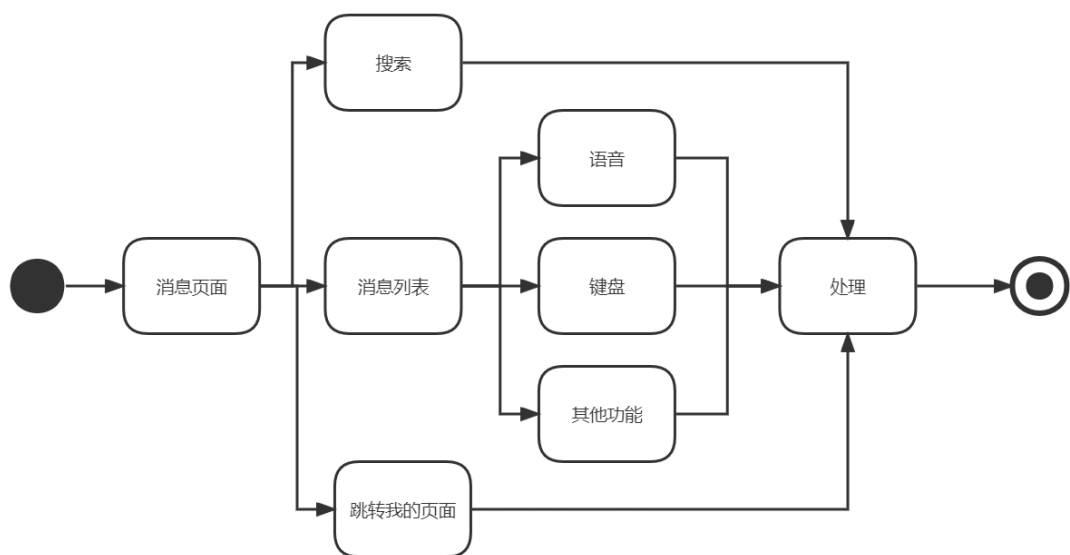


图 8 消息页面状态图

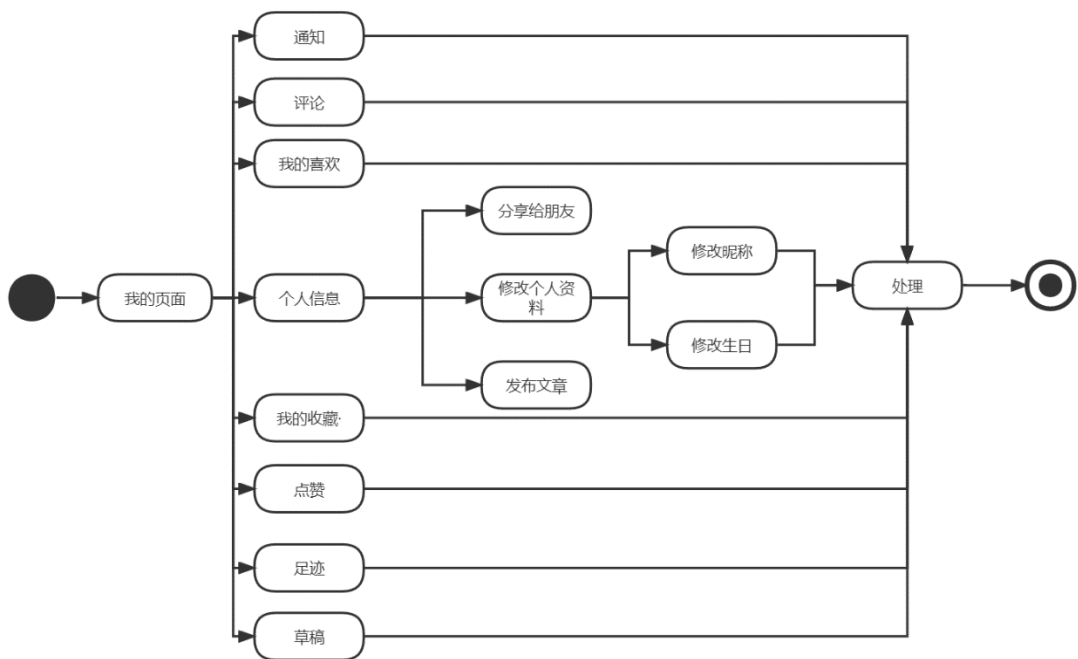


图 9 用户信息页面状态图

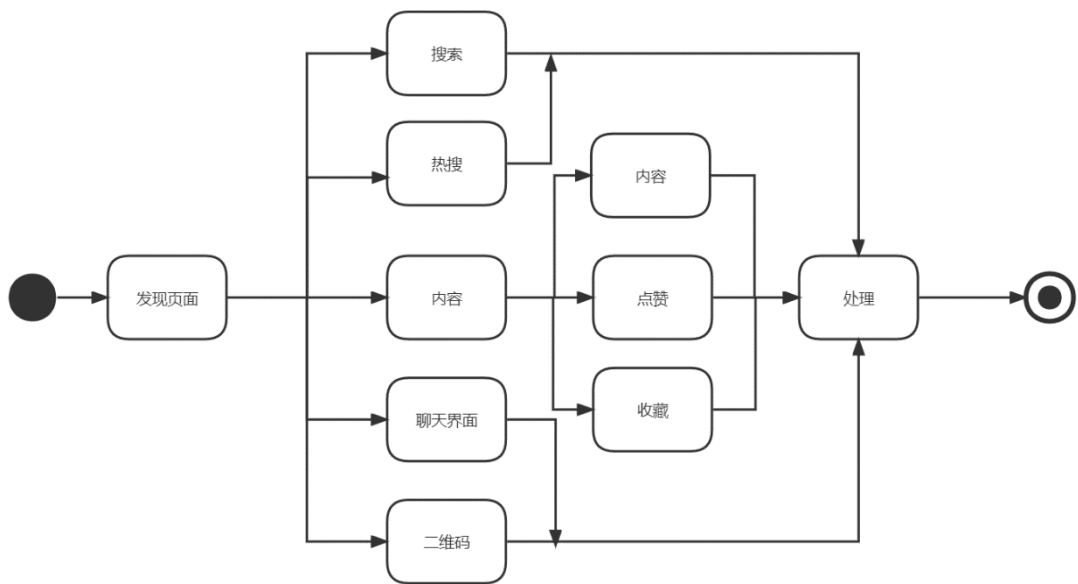
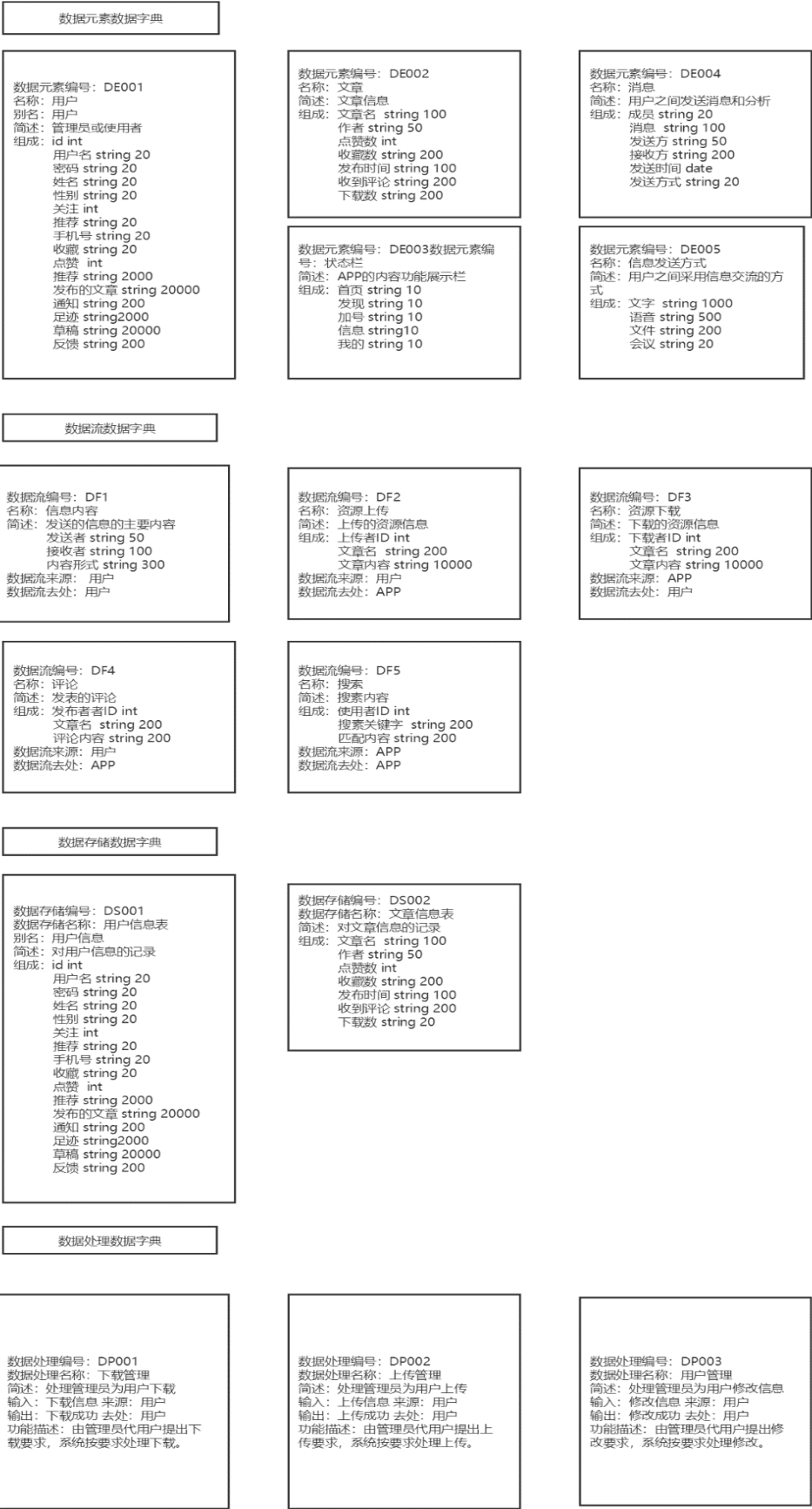


图 10 发现页面状态图

3.3 系统数据字典

软件系统数据字典如图 11:



3.4 数据库安全

数据库系统安全主要利用在系统级控制数据库的存取和使用的机制，包含：

- (1) 系统的安全设置及管理，包括法律法规、政策制度、实体安全等；
- (2) 数据库的访问控制和权限管理；
- (3) 用户的资源限制，包括访问、使用、存取、维护与管理等；
- (4) 系统运行安全及用户可执行的系统操作；
- (5) 数据库审计有效性；
- (6) 用户对象可用的磁盘空间及数量。

数据安全是在对象级控制数据库的访问、存取、加密、使用、应急处理和审计等机制，包括用户可存取指定的模式对象及在对象上允许作具体操作类型等。