

背包问题知识社区系统

详细设计说明书

文件状态： [<input checked="" type="checkbox"/>] 草稿 [<input type="checkbox"/>] 正式发布 [<input type="checkbox"/>] 正在修改	文件标识：	背包问题知识社区系统
	当前版本：	1.2
	作者：	杨凯
	完成日期：	2022/6/20

版本历史

版本/状态	作者	参与者	起止日期	备注
1.0	杨凯	杨凯、 阮凯、 潘晴、 孟姣姣	2022/5/14- 2022/5/16	开始编写，完成基本内容的设计
1.1	杨凯	杨凯、 阮凯、 潘晴、 孟姣姣	2022/6/5- 2022/6/10	确定软件采用的开发结构，对数据库进行进一步设计，设计出了数据库的逻辑结构
1.2	杨凯	杨凯、 阮凯、 潘晴、 孟姣姣	2022/6/18- 2022/6/20	细化了系统设计模型，精化了类的属性和操作，定义了服务参数和具体实现逻辑，调整类的层次关系，定义了软件数据库表的结构

修 改 记 录

日期	修订版本	修改章节	修改描述	作者
2022/6/7	V1.1	2.5、2.6	设计关键类的重点服务	杨凯
2022/6/20	V1.1	3.4	细化	杨凯

目录

1	引言	4
1.1	编写目的	4
1.2	背景	4
1.3	参考资料	4
2	设计概述	4
2.1	系统总体框架	4
2.2	系统子模块设计	5
2.3	系统模块功能	5
2.4	系统处理流程	6
2.5	设计关键类的重点服务	6
2.6	软件系统总体结构设计:	6
3	数据库系统设计	7
3.1	数据字典:	8
3.2	概念模型 (E-R图)	10
3.3	数据库逻辑结构 (E-R图向关系模型的转化)	10
3.4	数据库表结构:	11

1 引言

1.1 编写目的

系统设计文档编制的目的是说明一个软件系统各个层次中的每个程序（每个模块或子程序）和数据库系统的设计考虑，为程序员编码提供依据。

编写背包问题社区系统主要的目的是从实现需求的角度明确项目的各种功能模块，功能范围，明确与业务主机/终端系统的接口和业务处理模式，使系统开发人员和产品管理人员明确产品功能，可以有针对性的进行系统开发、测试、验收等各方面的工作。

1.2 背景

软件系统名称：背包问题知识社区 APP

该软件系统的基本功能：包含资源上传、检索、资源审核与管理、背包问题知识论坛。

系统的类型及从属地位：该系统主要面向对“背包问题”算法有需求，希望认识志趣相同的人的用户。属于知识提供类 APP。

开发项目组名称：夏日限定

1.3 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 设计概述

2.1 系统总体框架

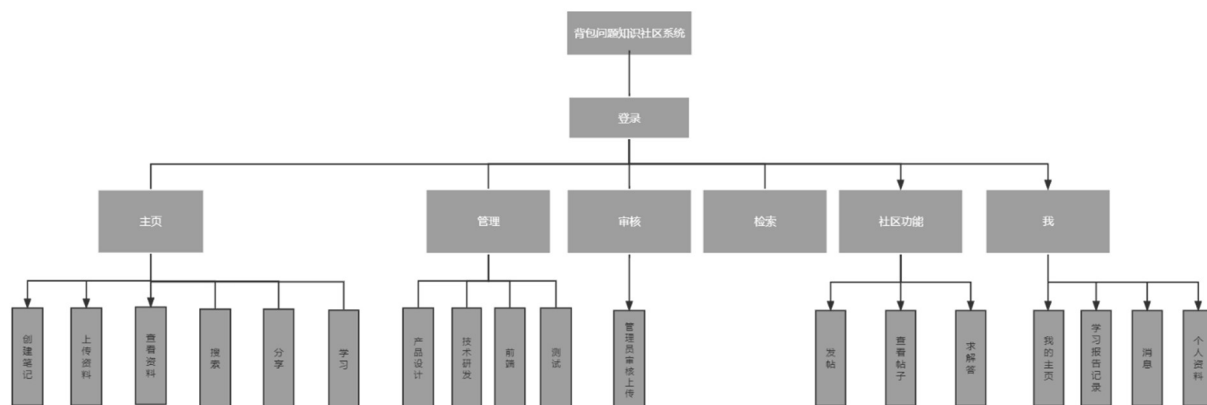


图2-1 功能层次图

2.2 系统子模块设计

0-1 背包问题知识社区系统的综合性相对较强，复杂程度相对较高，可以对软件进行充分利用，进行系统设计与规划。完整的背包问题知识社区系统，设计页面设计、处理程序、Mysql 后台数据库系统等，在页面将所有功能全部呈现。数据库方面需要存储用户信息，上传的算法资料，以及个人的发帖等等。

2.3 系统模块功能

- 1) 注册登录功能模块：用户通过手机号或者微信、微博、QQ进行注册登录，之后填写基本信息，通过注册后发送的雁验证码进入系统，之后可选择关注你喜欢的博主，进而进入软件。
- 2) 首页模块：显示学习打卡信息，日历以及进入学习选择界面。
- 3) 学习界面：进入学习界面，可选择学习的知识类型，此外，还有各种相关算法的代码详解。
- 4) 资料界面：包括资料上传以及检索功能，在上传资料后并不会立即显示在云端，管理员会对资料进行审核，并决定是否上传。
- 5) 社区功能：用户可以发帖，在“加油站”界面，可浏览多个用户发的贴，并进行讨论回复。
- 6) 我的个人信息界面：包含个人资料的编辑修改、查看、软件及开发团队的的基本信息。

2.4 系统处理流程

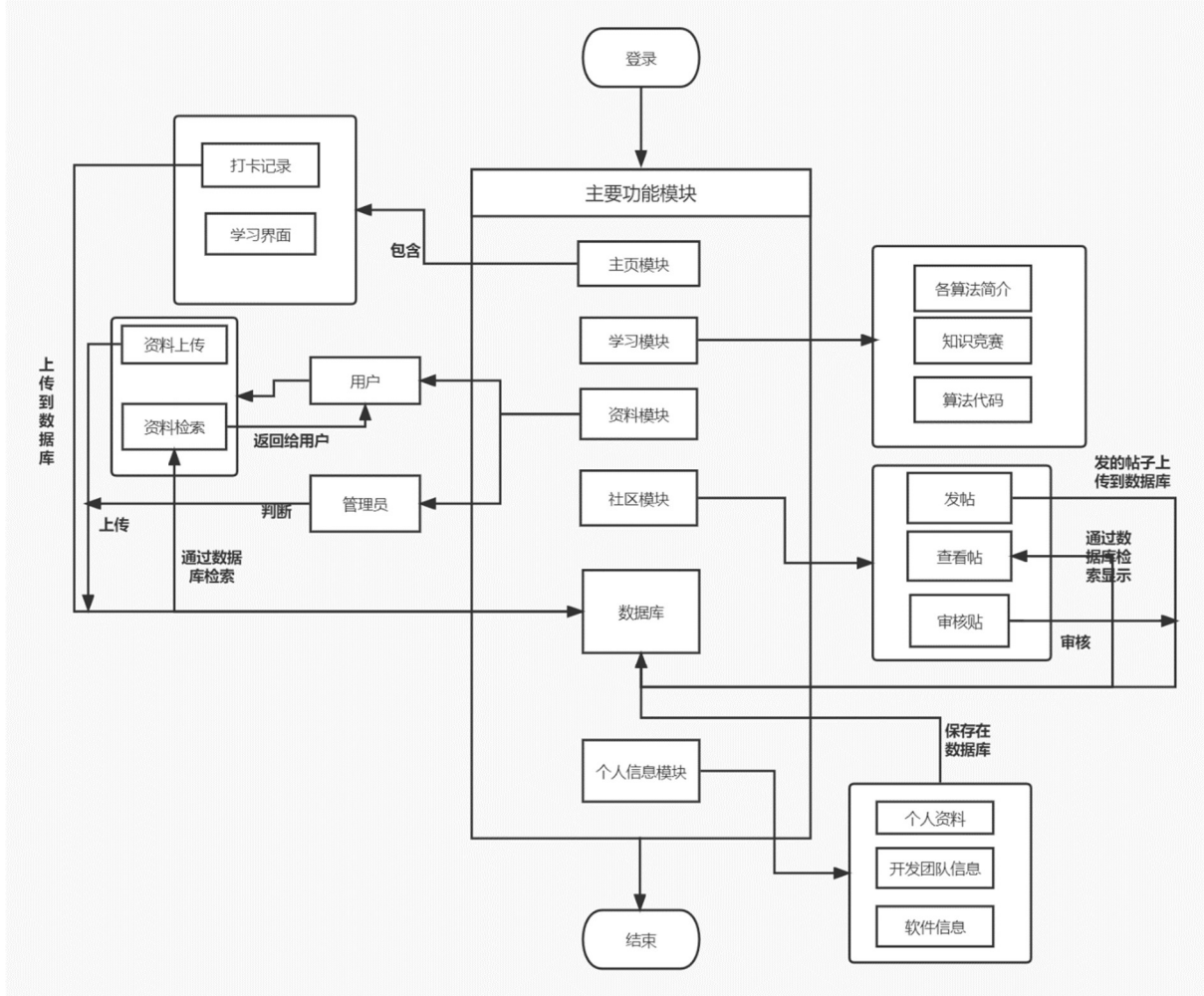


图2-2 系统处理流程图

2.5 设计关键类的重点服务

- User 类：包含普通用户以及管理员用户，用以表示用户信息
- dataBase 类：包含了数据库的连接，以及其增删改查的操作
- 用户功能类：包含了用户各个动作的函数
- 数据类：包含上传的资料、发表的评论、发表的帖子等基本类

2.6 软件系统总体结构设计：

本系统采用 Android 开发，若不适用构架进行开发，带来的问题是 Activity/Fragment 逻辑臃肿，不利于扩展。采用 MVC 设计模式，可以控制逻辑，数据处理逻辑和界面交互耦合。而 Android 开发中，其本身默认就可以理解为 MVC 结构，把 View 放在 xml 中与 Java 代码进行解耦，然后 Activity/Fragment 充当 Controller 进行逻辑控制，但是 Android 本身没有对 Model 进行划分，所

以在本系统中，我们可让 Activity/Fragment 充当 Model 和 Controller 两个角色。而且 xml 中的 View 操作也是在 Activity/Fragment 中，导致有时候 Activity/Fragment 也会充当一些 View 的角色。

此外本系统采用 Android，故默认为 C/S 软件体系结构。

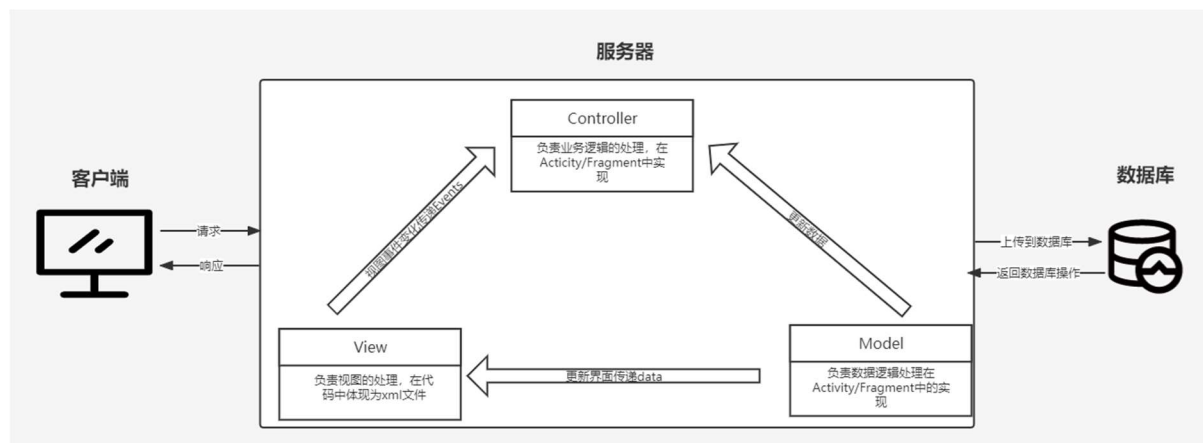


图2-3 软件系统结构

对软件系统的 MVC 设计模式进行详细设计：

< 1 >、**ControllerServlet.java**:请求分发控制器，解析请求中的动作名字，调用动作工厂对象生成的动作对象处理请求，处理后返回的结果为目的 URL，控制器再将请求和应答对象转向目标 URL

< 2 >、**ActionFactory.java**: 该工厂类将请求中的动作名转换成 servlet 可以用来完成其工作的动作类

< 3 >、**Action.java**: 接口类，该接口定义所有动作的公共接口

< 4 >、**具体 Action 的实现**: 指实现了 Action 接口的类，会被 ActionFactory 工厂根据请求中的名字创建，从而调用其具体实现的方法处理 Request\Response 对象后，返回一个 URL，由主控 servlet 转发给用户。可以理解为一个具体的 Action 实现类用于处理页面上每一种用户点击（请求）

3 数据库系统设计

数据库的设计关系到整个应用系统的运行效率，数据库设计的好，不仅有利于日常数据的维护更新，而且还可以提高系统的运行效率，缩短数据查询响应周期，增加网站的流量。合理的数据库设计可以使围绕它支持的页面的代码简单化，易于实现，并且可以提高数据存储的效率，保证数据的完整一致。背包问题知识社区系统采用 MySQL 作为后台数据库开发工具。

3.1 数据字典：

1. 用户：

字段名	数据类型	描述
user_id	VARCHAR(10)	用户编号
user_name	VARCHAR(10)	用户姓名
user_sex	VARCHAR(2)	用户性别
user_old	INT(3)	用户年龄
user_mail	VARCHAR(20)	用户邮箱
user_account	VARCHAR(20)	用户账号
user_password	VARCHAR(15)	登录密码

2. 资料：

字段名	数据类型	描述
info_id	VARCHAR(10)	资料编号
user_id	VARCHAR(10)	资料上传者的编号也就是用户编号
info_upload	VARCHAR(10)	资料上传者
info_title	VARCHAR(10)	资料名称
info_content	TEXT	资料内容
info_time	DATE	资料上传时间
info_update	DATE	资料更新时间
info_support	INT	资料点赞数
info_coamment	INT	资料评论数
info_comment_content	TEXT	资料评论内容
info_collection	INT	资料收藏数

3. 帖子：

字段名	数据类型	描述
post_id	VARCHAR(10)	帖子的编号

post_title	VARCHAR(10)	帖子的标题
post_date	DATE	发帖日期
user_name	VARCHAR(10)	发帖人的昵称也就是用户名称
user_id	VARCHAR(10)	发帖人的编号也就是用户编号
post_content	TEXT	帖子的内容
post_support	INT	帖子的点赞数
post_comment	INT	帖子的评论数
post_comment_content	TEXT	帖子的评论内容
post_collection	INT	帖子的收藏数

4. 评论者：

字段名	数据类型	描述
comment_id	VARCHAR(10)	评论编号
user_id	VARCHAR(10)	评论者的编号也就是用户编号
user_name	VARCHAR(10)	评论者的昵称也就是用户名称
comment_date	VARCHAR(10)	评论时间
commented_ida	VARCHAR(10)	被评论者编号
commented_name	VARCHAR(10)	被评论者昵称

3.2 概念模型（E-R 图）

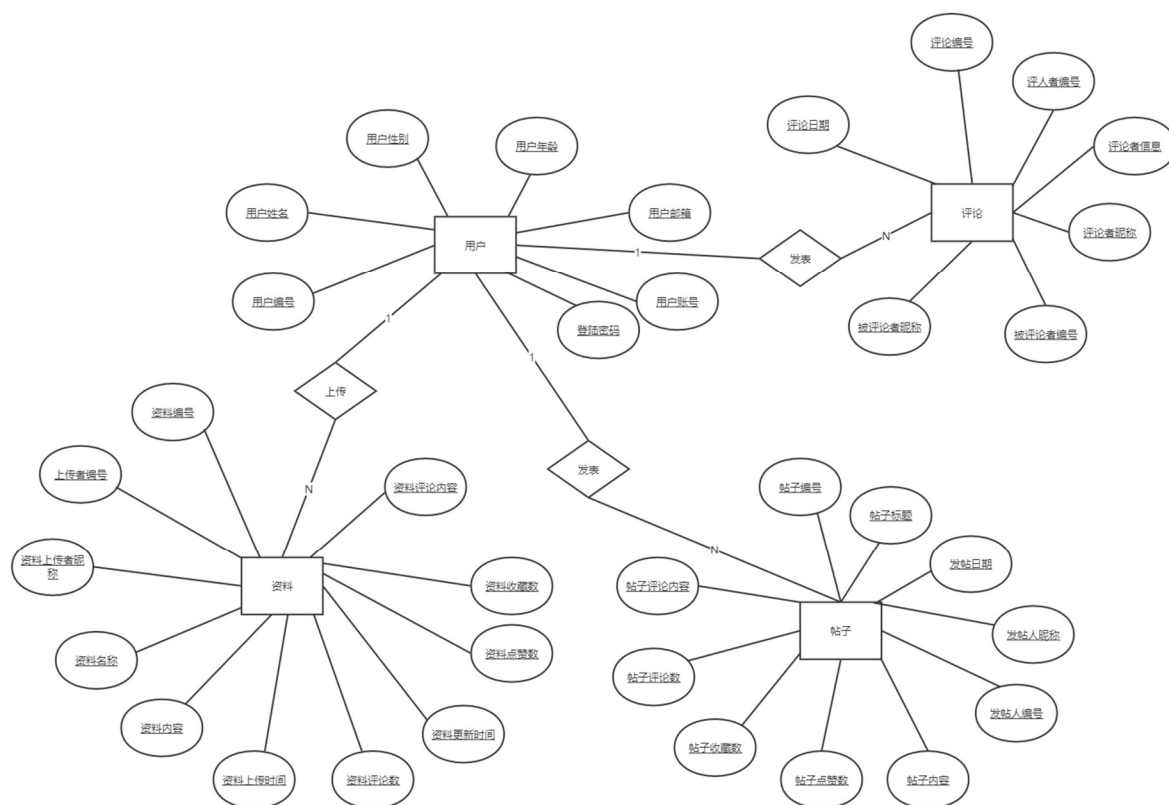


图3-1 ER图

3.3 数据库逻辑结构（E-R 图向关系模型的转化）

• 用户-资料：

用户（*用户编号*，用户姓名，用户性别，用户年龄，用户邮箱，用户账号，登陆密码）

资料（*资料编号*，上传者编号，资料上传者昵称，资料名称，资料名称，资料内容，资料上传时间，资料评论数，资料更新时间，资料点赞数，资料收藏数，资料评论内容）

上传（用户编号，资料编号）

• 用户-评论

用户（*用户编号*，用户姓名，用户性别，用户年龄，用户邮箱，用户账号，登陆密码）

评论（*评论编号*，评论者编号，评论者昵称，被评论者编号，被评论者昵称）

发表（用户编号，评论）

- 用户-帖子

用户（用户编号，用户姓名，用户性别，用户年龄，用户邮箱，用户账号，登陆密码）

帖子（帖子编号，帖子标题，发帖日期，发帖人昵称，发帖人编号，帖子内容，帖子点赞数，帖子收藏数，帖子评论数，帖子评论内容）

发表（用户编号，帖子编号）

3.4 数据库表结构：

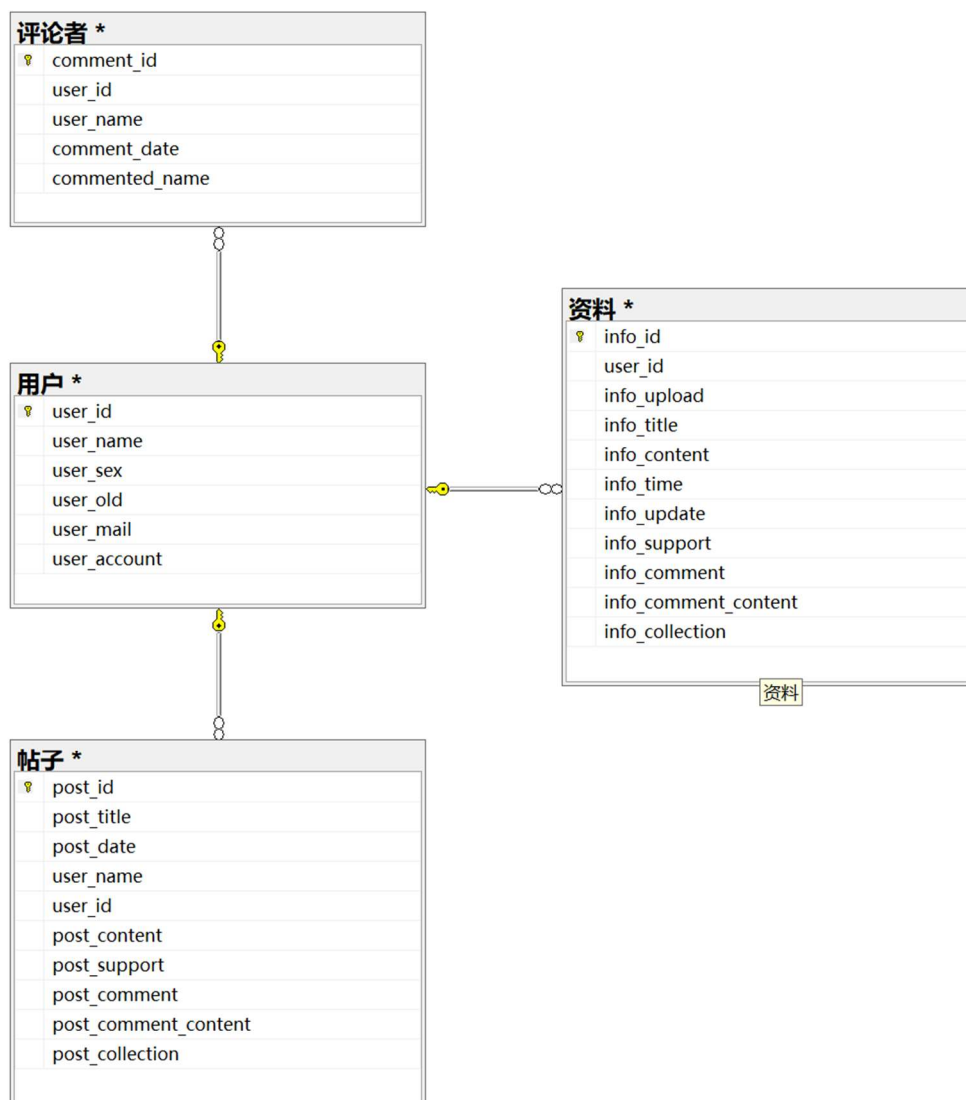


图3-2 数据库表结构