背包问题知识社区

系统设计说明书

团队名称：无霸哥

团队编号：202214

负 责 人：曹霖枫

指导教师：代祖华

团队成员：牛靖威 王孜睿

目录

第一部分 项目概述

1.1 项目背景

背包问题(Knapsack problem)是一种组合优化的NP完全问题。问题可以描述为：给定一组物品，每种物品都有自己的重量和价格，在限定的总重量内，我们如何选择，才能使得物品的总价格最高。问题的名称来源于如何选择最合适的物品放置于给定背包中。相似问题经常出现在商业、组合数学，计算复杂性理论、密码学和应用数学等领域中。也可以将背包问题描述为决定性问题，即在总重量不超过W的前提下，总价值是否能达到V？它是在1978年由Merkle和Hellman提出的。在计算机科学与技术学科教学中十分常见。

1.2 研究目的

提供学习背包问题的平台。

在背包问题的学习中作为工具为用户答疑解惑，提升解决问题能力。

2.系统需求分析

2.1 性能需求分析

本系统为由注册用户共享的背包问题知识社区系统，此类系统在性能上需要稳定、高效和可靠;在结构上应该具有很好的可扩展性，便于将来功能的扩展和维护。前端采用HTML、CSS、jQuery，数据库采用Mysql5.7，使用墨刀进行设计，方便简洁。

2.2 用户需求分析

本系统为由注册用户共享的背包问题知识社区系统。

系统具有资源上传、检索、资源审核与管理、背包问题主题知识论坛。方便

用户沟通交流。

资源类型至少包括：典型算法源代码、开源数据集、背包问题相关文献资源。

系统设置管理员，用于系统用户管理、资源有效性审核。

UI 美观合理，方便使用。

图 2.2.1 四个象限

第二部分 软件系统总体结构

3.1 系统结构

MVC 全名是 Model View Controller，是模型(model)－视图(view)－控制器(controller)的缩写，一种软件设计典范，用一种业务逻辑、数据、界面显示分离的方法组织代码，将业务逻辑聚集到一个部件里面，在改进和个性化定制界面及用户交互的同时，不需要重新编写业务逻辑。MVC 被独特的发展起来用于映射传统的输入、处理和输出功能在一个逻辑的图形化用户界面的结构中。

Model（模型）是应用程序中用于处理应用程序数据逻辑的部分。

通常模型对象负责在数据库中存取数据。

View（视图）是应用程序中处理数据显示的部分。

通常视图是依据模型数据创建的。

Controller（控制器）是应用程序中处理用户交互的部分。

通常控制器负责从视图读取数据，控制用户输入，并向模型发送数据。

MVC 分层有助于管理复杂的应用程序，因为您可以在一个时间内专门关注一个方面。例如，您可以在不依赖业务逻辑的情况下专注于视图设计。同时也让应用程序的测试更加容易。

B/S 结构（Browser/Server，浏览器/服务器模式），是 WEB 兴起后的一种网络结构模式，WEB 浏览器是客户端最主要的应用软件。这种模式统一了客户端，将系统功能实现的核心部分集中到服务器上，简化了系统的开发、维护和使用。客户机上只要安装一个浏览器，如 Netscape Navigator 或 Internet Explorer，服务器安装 SQL Server、Oracle、MYSQL 等数据库。浏览器通过 Web Server 同数据库进行数据交互。

第一层是浏览器，即客户端，只有简单的输入输出功能，处理极少部分的事务逻辑。由于客户不需要安装客户端，只要有浏览器就能上网浏览，所以它面向的是大范围的用户，所以界面设计得比较简单，通用。

第二层是 WEB 服务器，扮演着信息传送的角色。当用户想要访问数据库时，就会首先向 WEB 服务器发送请求，WEB 服务器统一请求后会向数据库服务器发送访问数据库的请求，这个请求是以 SQL 语句实现的。

第三层是数据库服务器，他扮演着重要的角色，因为它存放着大量的数据。当数据库服务器收到了 WEB 服务器的请求后，会对 SQL 语句进行处理，并将返回的结果发送给 WEB 服务器，接下来，WEB 服务器将收到的数据结果转换为 HTML 文本形式发送给浏览器，也就是我们打开浏览器看到的界面。

在整体设计中，我们将宿舍管理系统分为六个大的模块：系统管理模块、宿管系统、学籍信息系统、后勤管理系统、宿舍信息系统、关于模块。每个模块将实现不同的功能。下面将具体进行介绍。

3.2 系统管理模块

系统管理模块包括：超级用户登录、学生用户登录、用户管理、退出系统五

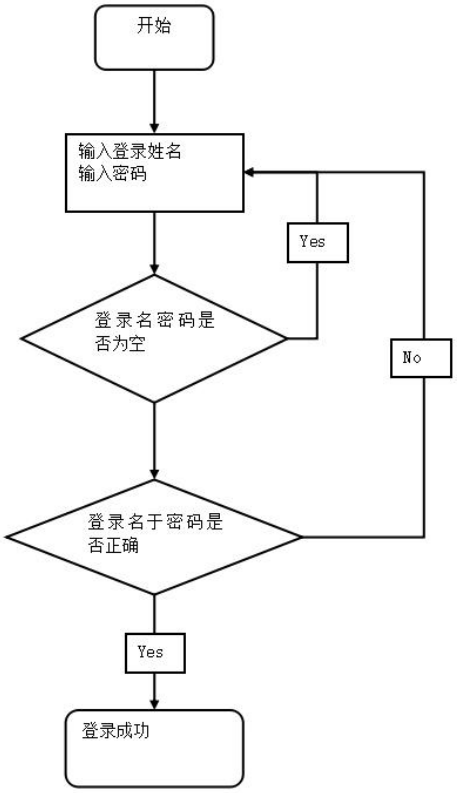
部分。

1.超级用户管理：实现超级管理员的登录，可以进行各级用户权限管理，创建新用户等功能。

3.学生用户登录：实现学生用户登录，可以进行数据查询，发帖提问等功能。

4.用户管理：实现各级用户的自我管理，可以进行查看基本信息等功能，管理人员授予或取消一般用户登录该系统的用户名和密码。

5.退出系统：实现正常退出系统



3.3 关于系统模块

本系统分为一下几个模块

1. 用户注册登录模块；
2. 用户管理模块（管理员权限）；
3. 文章内容浏览模块；
4. 点赞评论收藏交互模块；
5. 个人内容管理模块。

第四部分 数据库设计概述

4.1 数据库环境说明

数据库环境为 SQL Server 8.0

数据库服务器: Oracle/MySql，能够处理数据并发访问，访问回馈时间短。

4.2 数据库概念结构

4.2.1 各实体属性

管理员：管理员编号，姓名

用户：编号，姓名，性别

专业人员：编号，姓名

4.2.2 E-R 模型

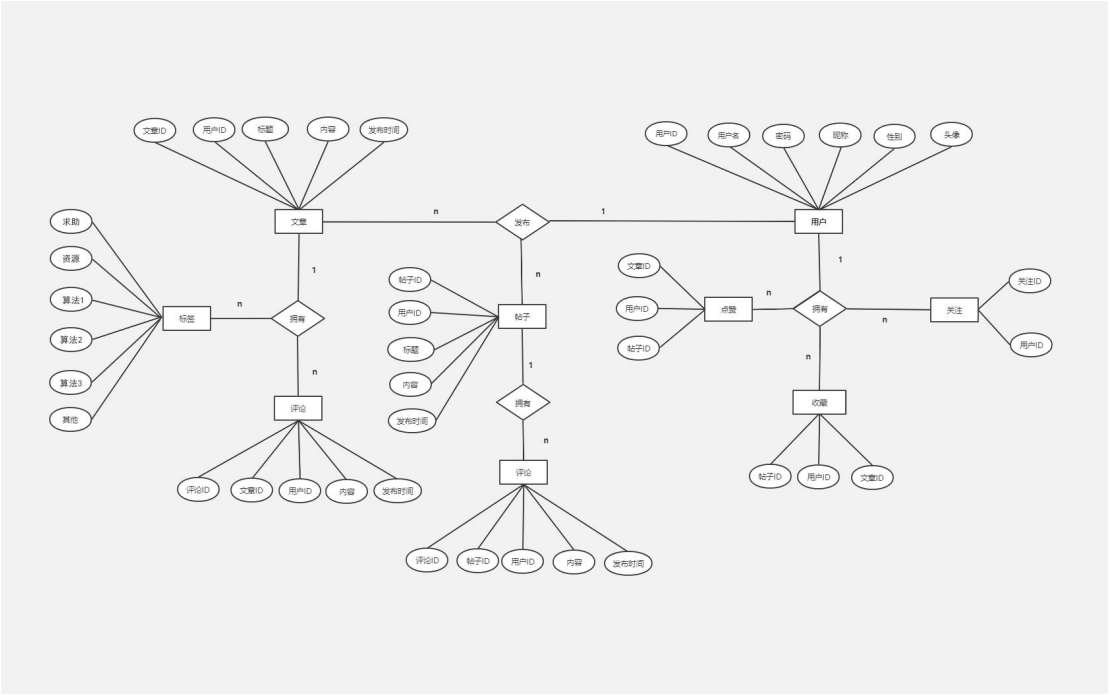


图 6-1 E-R 图

4.3 逻辑结构——关系模型

4.3.1 E-R 图向关系模型的转换

4.3.2 数据字典

数据项描述={数据项编号，数据项名，数据项含义说明，数据类型，长度}。

数据	说明	类型和长度	约束

Uname	用户名	Varchar(20)	非空
Uid	用户 ID	Int(10)	非空
Upasswd	用户密码	Varchar(20)	非空
Usex	用户性别	Varchar(20)	非空
Aname	管理员名	Varchar(20)	非空
Aid	管理员 ID	Int(10)	非空
Apasswd	管理员密码	Varchar(20)	非空
Tid	帖子 ID	Int(10)	非空
Tclass	帖子类别	Varchar(20)	非空

4.4 物理结构

4.4.1 物理设计的内容和方法

为了设计优化的物理数据库结构，使得在数据库上运行的各种事务响应时间小、存储空间利用率高、事务吞吐量达。为此，首先对要运行的事务进行详细分析，其次，要充分了解所用关系数据库管理系统的内部特征。

被更新的关系，用户信息关系上的更新操作条件所涉及的属性，修改操作要改变的属性值，对每个事务在各关系上运行的频率和性能尽可能达到要求。

通过以上信息来确定关系的存取方法。

4.4.2 数据库的存储结构

关系模型的存储安排：

学生信息表　存储学生的基本信息

管理员表　　存储所有管理员的基本信息

数据库物理结构的内容：关系，索引，聚簇，日志，备份。

4.4.2.1 确定数据的存放位置

数据库数据备份，日志文件备份等由于只在故障恢复时才使用，而且数据量很大，可以考虑放在磁盘上；而且将表和索引分别放在不同的磁盘上，在查询时，由于两个磁盘驱动器分别在工作，因而可以保证物理读写速度与较快；

此外可以将不较大的表分别存放在两个磁盘上，以加快存取速度，这在多用户条件下特别有效。

4.4.2.2 确定系统配置

系统都为这些变量赋予了合理的缺省值。但是这些值不一定适合每一种应用环境，在进行物理设计时，需要根据应用环境确定这些参数，以使系统性能最优。

在物理设计时对系统配置变量的调整坐在这只是初步的，在系统运行时还要根据系统数据运行情况作进一步调整，以期切实改进系统性能。

第五部分 flask 开发环境

5.1 虚拟环境

虚拟环境是 python 解释器的一个私有副本，在这个环境中你可以安装私有包，而且不会影响系统中安装的全局 python 解释器。这样做的好处：

为每个项目单独创建虚拟环境，可以保证应用只能访问所在虚拟环境中的包，从而保证全局解释器的干净整洁，使其只作为创建更多虚拟环境的源；

使用虚拟环境不需要管理员权限

5.2 pip 工作流

创建虚拟环境：python3 和 python2 创建虚拟环境的方法有所不同:

python3 中虚拟环境由标准库中的 ven 包原生支持：python3 -m venv venv(虚拟环境的名称)。通常虚拟环境的名称为 venv，上述命令执行完毕后，项目目录中会出现一个名为 venv 的子目录，这里就是一个全新的虚拟环境，包含这个项目专用的 Python 解释器。

python2 没有集成 venv 包，这一版 python 解释器要使用第三方工具 virtualenv 创建虚拟环境：sudo pip install virtualenv + virtualenv venv;

激活虚拟环境：

source venv/bin/active(Linux) + venv\Scripts\active(windows)

输入 deactive 退出虚拟环境，即还原当前终端会话的 PATH 环境变量。

使用 pip 安装包：

python 包使用包管理器 pip 安装，所有虚拟环境中都有这个工具

使用豆瓣镜像安装：pip install flask -i https://pypi.douban.com/simple

pip freeze > requirements.txt 生成需求文件

pip install -r requirements.txt 安装 requirements.txt 列出的依赖包

5.3 Pipenv 工作流

Pipenv 是基于 pip 的 python 包管理工具，它的用法与 pip 相似，可以看作是 pip 的加强版，它的出现解决了旧的 pip + virtualenv + requirements.txt 的工作方式的弊端，使得包安装、包依赖管理和虚拟环境管理更加方便：

安装 Pipenv：pip install pipenv -i https://pypi.douban.com/simple

执行 pipenv install：会为当前项目创建一个文件夹，该文件夹包含

隔离的 python 解释器环境，并安装 pip、wheel、setuptools 等基本包；

如果项目中包含 Pipfile 文件，其中的依赖包也会被一并安装。

默认情况下 pipenv 会同一管理所有虚拟环境，虚拟环境文件夹会在在 C：\Users\Administrator\.virtualenvs\（Linux：~/.local/share/virtualenvs/）。若想在项目目录内创建虚拟环境文件夹，可以添加环境变量 WORKON_HOME=PIPENV_VENV_IN_PROJECT。

pipenv shell：激活虚拟环境

pipenv run python hello.py：使用当前项目的虚拟环境运行 hello.py 程序。

5.4 总结

在以前我们通常使用 pip 搭配一个 requirements.txt 文件来记录 依赖。但 requirements.txt 需要手动维护，在使用上不够灵活。Pipfile 的出 现就是为了替代难于管理的 requirements.txt。

在创建虚拟环境时，如果项目根目录下没有 Pipfile 文件，pipenv install 命令

还会在项目文件夹根目录下创建 Pipfile 和 Pipfile.lock 文件，前 者用来记录项目依赖包列表，而后者记录了固定版本的详细依赖包列 表。当我们使用 Pipenv 安装/删除/更新依赖包时，Pipfile 以及 Pipfile.lock 会自动更新。

当需要在一个新的环境运行程序时，只需要执行 pipenv install 命令。Pipenv 就会创建一个新的虚拟环境，然后自动从 Pipfile 中读取依赖并安装到新创建的虚拟环境中。

Pipenv 会自动帮我们管理虚拟环境，所以在执行 pipenv install 安装 Python 包时，无论是否激活虚拟环境，包都会安装到虚拟环境中。后面 我们都将使用 Pipenv 安装包，这相当于在激活虚拟环境的情况下使用 pip 安装包。只有需要在全局环境下安装/更新/删除包，我们才会使用 pip。

5.5 Flask 依赖包

pipenv install flask 后，除了 Flask 包外，同时被安装的还有 5 个依赖包：

Jinjia2：模板渲染引擎

MarkupSafe：HTML 字符转义工具

Werkzeug：WSGI 工具集，处理请求与响应、内置 WSGI 开发服务器、调试器和重载器；

click：命令行工具

itsdangerous：提供各种加密签名功能

第六部分 软件重用概述

5.1 知识重用方案

知识如此珍贵，最好能重复使用：不仅自己使用，也让别人使用。让自己人不用花代价，就可以获得知识。于是，知识传播也变得重要起来。人类的祖先“智人”之所以能够战胜“尼人”，一个重要的原因就是“智人”可以用语言传播知识。

设备远程维护最好让设备开发商参与，就是因为他们便于知识复用。我们国家制造业的优势在于市场大——市场大的好处就是便于知识的重用。在有些行业中，企业越来越大。其中重要的“粘合剂”就是知识。

一般来说，知识需要在使用的过程中才能不断丰富、完善。这就叫“从实践中来，到实践中去”。实践多了，才能走向成熟。知识成熟了，使用的风险就小。所以，阿波罗计划拒绝采用不成熟的技术。进入现代工业社会以后，企业都要求按照标准化进行生产。而“标准”就是一种成熟以后固化起来的知识。在智能化的时代，知识可以固化在计算机里，自动地使用知识；可以在互联网上传播，极大地促进知识的重用。知识被重用的次数多了，获得知识的成本就可以被摊平，从而进一步促进知识的产生。而大数据的时代，能够帮助人类方便地获得更多的知识——甚至包括图像识别这样不容易描述的“感性知识”。所以，认识智能化时代的一个角度，是知识经济。

对于本系统，学习过的知识可以得到充分利用，比如数据库，JAVAEE，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。

5.2 方法和标准的重用方案

对于本系统，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。

5.3 软件成分的重用方案

软件重用分类比较困难，因为软件重用技术众多，一种重用技术可以包括多种重用形式。比如说：框架即可以包括代码级重用，也可以包括设计级重用。有一种分类方法是按照软件重用所应用的领域范围，把重用划分为两种：横向重用和纵向重用。

1.横向重用是指重用不同应用领域中的软件元素，例如数据结构、分类算法、人机界面构件等。标准函数库是一种典型的原始的横向重用机制。

2.纵向重用是指在一类具有较多公共性的应用领域之间进行软部品重用。因为在两个截然不同的应用领域之间实施软件重用非常困难，潜力不大，所以纵向

重用才广受瞩目，并成为软件重用技术的真正所在。纵向重用活动的主要包括以下几个步骤：

1) 首先进行域分析。根据应用领域的特征及相似性预测软部件的可重用性。

2) 然后进行软部品的开发。一旦确认了软部件的重用价值，即可进行软部品的开发并对具有重用价值的软部品进行一般化，以便它们能够适应新的类似的应用领域。

3) 最后，软部件及其文档即可进入软部品库，成为可供后续项目使用的可重用资源。

在本系统中，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。

5.4 类构件实现软件重用方案设计

利用面向对象技术,可以更方便更有效地实现软件重用。面向对象技术中的“类”,是比较理想的可重用软构件,不妨称之为类构件。类构件有 3 种重用方式,分别是实例重用、继承重用和多态重用。下面进一步讲述与类构件有关的内容。

类构件的重用方式

（1）实例重用

由于类的封装性，使用者无须了解实现细节就可以使用适当的构造函数，按照需要创建类的实例，然后向所创建的实例发送适当的消息，启动相应的服务,完成需要完成的工作，这是最基本的重用方式。此外，还可以用几个简单的对象作为类的成员创建出一个更复杂的类,这是实例重用的另一种形式。

虽然实例重用是最基本的重用方式,但是,设计出一个理想的类构件并不是一件容易的事情。例如，决定一个类对外提供多少服务就是一件相当困难的事，提

供的服务过多会增加接口复杂度，也会使类构件变得难于理解；提供的服务过少,则会因为过分一般化失去重用价值。每个类构件的合理服务数都与具体应用环境密切相关，因此找到一个合理的折衷值是相当困难的。

（2）继承重用

面向对象方法特有的继承性提供了一种对已有的类构件进行裁剪的机制,当已有的类构件不能通过实例重用完全满足当前系统需求时,继承重用提供了一种安全地修改已有类构件,以便在当前系统中重用的手段。

为提高承重用的效果,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。这样做有下述两个好处:

1）每个子类在继承父类的属性和服务的基础上,只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,面且为软件开发人员提供了更多可重用的类构件。因此,在软件开发过程中,应该时刻注意提取这种潜在的可重用构件,必要时应在领域专家帮助下,建立符合领域知识的继承层次。

2）为多态重用奠定良好基础

（3）多态重用

利用多态性不仅可以使对象的对外接口更加一般化(基类与派生类的许多对外接口是相同的),从而降低了消息连接的复杂程度,而且还提供了一种简便可靠的软构件组合机制,系统运行时,根据接收消息的对象类型,由多态性机制启动正确的方法,去响应一个一般化的清息,从而简化了消息界面和软构件连接过程。

对于本系统我们使用继承重用，对于面向对象语言来说，必须要保证它的可封装性，就要用到继承机制来达到代码复用的目的。对于接口重用，我们考虑用类指针调用派生类的方法来实现接口重用。比如在后期的编码中，学籍信息类可以学生类。

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。

第六部分 关键类的服务重点

本系统根据状态图和数据流图可筛选出本项目的关键类及相应重点服务如下：

管理人员：根据审核发布的帖子，为用户进行信息反馈，进行相关内容的推送。

用户：可以进行数据查询，发送帖子，进行相关信息的寻找。