



搜索网站项目^①

网盘作为一种备受欢迎的在线文件共享方式,被广泛运用于存储各类软件、游戏、视频、音乐、电子书等资源。这些由千千万万网民上传的内容组成了一个庞大的资源宝库。然而,网盘服务网站通常不提供检索功能,同时一些通用网络搜索引擎,如 Google、百度等也没有专门对网盘资源进行索引。因此,检索效果往往不尽如人意,用户常常面临信息分散、质量参差不齐、内容陈旧过时等问题。目前,网络上的网盘搜索引擎相对较少,这进一步限制了用户对于特定资源的搜索和获取。

基于上述背景,本项目开发了一个集成的资源平台,旨在为普通用户提供便捷的互联网资源搜索的途径。该平台使用户能够轻松地搜索并下载各种形式的互联网资源,涵盖文档、视频、音频、图片等多种类型。

5.1 相关背景

根据中国互联网络信息中心(CNNIC)发布的数据,截至 2016 年底,中国网民规模约为 7.31 亿人,并预计未来会持续大幅增长。这反映了互联网在中国的普及率日益提高,人们对网络资源的需求也随之增加。随着信息技术和社交网络的快速发展,互联网产生的数据量在近年来迅速增长,标志着大数据时代的来临。在大数据时代的背景下,互联网的快速发展不仅使得网民规模不断增长,还导致了互联网上产生的海量数据急剧增加。这巨大的数据量包含了各类信息,从而使得用户在海量数据中找到特定内容变得更加困难。网盘作为一种便捷的在线文件共享方式,成为许多人存储和分享个人及社区资源的首选。然而,由于通用搜索引擎对网盘资源的检索能力有限,用户往往陷入信息过载,难以精准获取所

^① 本案例由黄义炽、袁炳杰和章明亮(来自福州大学 2016 级软件工程专业)提供。

需的资源。

为了解决这一问题,本项目引入了网络爬虫技术。网络爬虫能够按照一定规则自动抓取互联网上的信息,并收录到系统数据库中,为用户提供更便捷的资源搜索途径。通过设计高效的爬虫程序,并结合自有搜索技术,我们开发出一个资源搜索网站,实现了对互联网上各种形式资源的全面抓取和整理。这个网站不仅提供了用户高效搜索的功能,还能够通过自有的爬虫程序不断更新数据库,确保用户获取的搜索结果具有较高的相关度。此外,该网站还提供了更加干净、美观的搜索界面。

通过这样的系统,用户可以方便地搜索和下载互联网上的各类资源。网站的用户群体覆盖了所有在互联网上搜索和下载网盘资源的普通用户。

总体而言,这样的网盘搜索引擎不仅能够满足用户对各类资源的需求,还在大数据时代的信息爆炸中发挥了关键作用,为用户提供了更加便捷、高效和个性化的资源搜索和下载体验。

5.2 需求分析

5.2.1 用例图

爬虫员能够选择不同的爬虫方式,以满足特定需求并优化资源抓取的效果。网络爬虫员的用例图如图 5-1 所示。下面介绍本案例网络爬虫员的用例图。

爬取网盘地址:爬虫员通过这个子用例可以选择爬取特定网盘地址中的资源信息。这包括了从特定网盘中检索所有资源的需求,以确保系统中包含了广泛的网盘资源。

爬取资源主题:爬虫员通过这个子用例能够按照资源的主题或关键词进行抓取。这使得系统可以囊括各种主题,满足用户对不同内容的兴趣。

爬取资源分类:该子用例允许爬虫员按照特定的分类方式抓取资源,确保系统中的资源按照清晰的分类体系组织。

爬取资源大小:爬虫员可以选择按照资源的大小范围进行爬取。这确保了系统中包含不同大小的资源,满足用户对文件大小的不同需求。

爬取上传时间:爬虫员能够抓取最新上传的资源,保持系统中的资源信息的时效性。

存入数据库:爬虫员通过所选的爬虫方式获得的资源信息将被存入系统的数据库中。这包括将网盘地址、资源主题、资源分类、资源大小、上传时间等信息整合并存储,以便系统

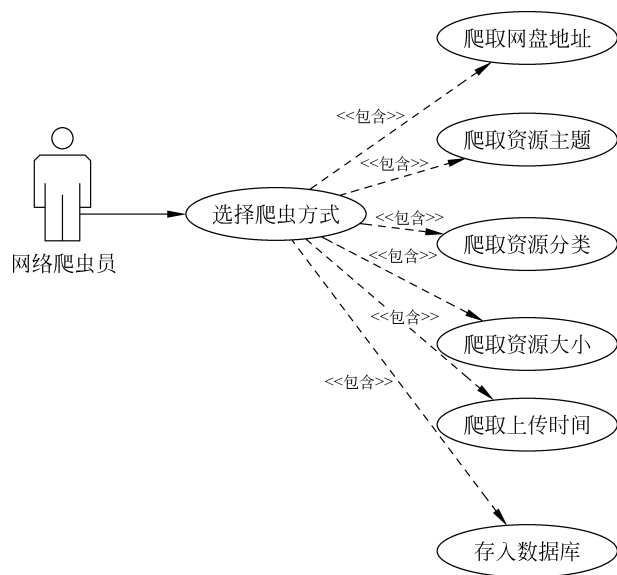


图 5-1 网络爬虫员用例图

能够快速、准确地提供搜索和浏览服务。

通过这些灵活的爬虫方式,系统的资源库能够更全面、深入地满足用户的需求,同时保持数据库的实时和完整。

用户用例图如图 5-2 所示。下面根据图 5-2 进行具体的介绍。

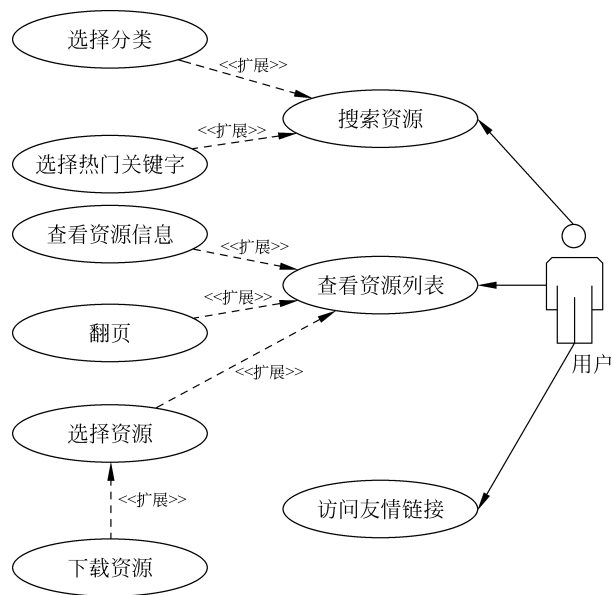


图 5-2 用户用例图

搜索资源：用户可以通过输入关键词实现对资源的搜索。在搜索过程中，用户可以使用以下拓展选项：选择分类、选择热门关键词。其中，选择分类是指用户可以在搜索框的旁边选择特定的资源分类，以便更准确地定位感兴趣的内容；选择热门关键词是指用户可以单击系统提供的热门关键词，以便快速获取当前热门或相关的资源信息。

查看资源列表：用户在搜索后会得到一个资源列表。这一用例可以进一步拓展成查看资源信息、翻页和选择资源。查看资源信息是指用户可以通过单击列表中的资源项来查看详细信息，包括文件大小、上传者、上传时间等。翻页是指如果搜索结果超过一页，用户可以通过翻页按钮或滚动浏览更多资源，确保他们能够访问到系统中的全部资源。选择资源是指用户可以通过勾选资源的复选框或其他方式选择他们感兴趣的资源，以备后续的操作。

访问友情链接：用户可以单击系统提供的友情链接，这可能包括与资源相关的其他网站、合作伙伴站点等，以便获取更多相关信息。

下载资源：用户可以选择并下载感兴趣的资源。可以具体拓展成前文所提到的选择资源。

通过这些细化的用例，系统可以更精确地了解用户在每个操作步骤中的需求，从而提供更加智能、用户友好的交互体验。

后台管理员用例图如图 5-3 所示。后台管理员在登录之后的操作主要有修改密码、查看网页和查看数据库，下面根据这几部分详细介绍本案例后台管理员的用例图。

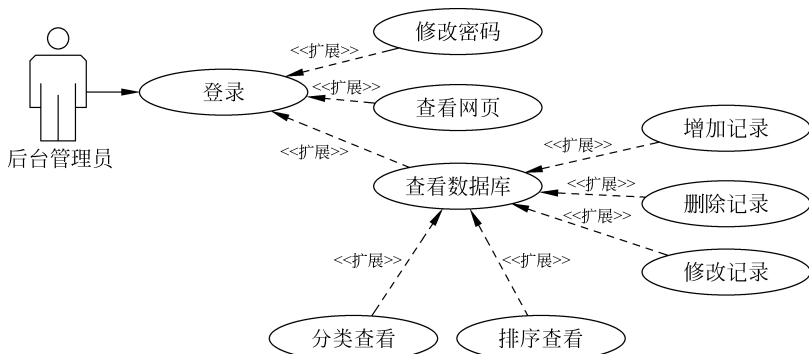


图 5-3 后台管理员用例图

修改密码：后台管理员可以通过此功能修改其登录系统的密码，确保账户安全。

查看网页：后台管理员可以通过此功能查看系统的前端页面，以便了解系统的用户界面和交互设计。

查看数据库：主要包括增加、删除、修改记录，以及分类、排序查看这几部分的子功能。后台管理员可以对数据库中的记录进行增加、删除和修改操作，以维护和管理系统的数据。

同时,后台管理员可以按照不同的分类标准查看数据库中的记录,以便更好地组织和管理数据;按照指定的字段对数据库中的记录进行排序,以便更方便地查找和浏览数据。

这些功能使得后台管理员能够对系统的数据进行灵活管理和维护,确保系统的正常运行和数据的完整性。

5.2.2 原型图

下面以后台站点管理界面为例介绍本项目的原型图设计。

如图 5-4 所示,后台站点管理界面中,左侧主要是新建用户或新建组、用户列表、网盘资源、资源列表几个功能;右侧是“最近动作”和“我的动作”列表,可以帮助后台管理员查看信息,方便管理资源站点和用户。界面上的最近动作类似于历史操作,可以帮助用户轻松追溯和跟踪最近的操作记录,包括对用户、组、资源等的创建、编辑、删除等动作。而“我的动作”列表则是针对当前登录的后台管理员所执行的操作,以便管理员快速了解自己的操作历史和活动。整个后台站点管理界面设计简洁清晰,功能布局合理,使得管理员能够高效地管理用户、资源和站点信息。



图 5-4 后台站点管理界面

5.3 系统设计

5.3.1 体系结构设计

在项目的开发过程中,选择采用 Python 作为主要开发语言。Python 以其清晰简洁的语法和广泛的应用领域,使得团队成员能够高效协作,极大地缩短了开发周期。同时,Python 拥有丰富的生态系统和强大的库支持,为项目提供了丰富的工具和解决方案,使得开发过程更加顺利和高效。

在页面设计方面,本项目采用了 CSS 和 JavaScript 这两种前端技术。CSS 主要用于页面样式设计,利用 DIV+CSS 实现了页面的布局。这种设计模式不仅使页面呈现更为美观,而且充分体现了表现与内容的分离原则。这种分离不仅有助于提高网页加载速度,还使得网页的维护和改版变得更为便利。

JavaScript 作为一种客户端脚本语言,主要在 Web 浏览器中解释执行,为网页增加了丰富的交互和动态效果。通过 JavaScript,本项目成功实现了行为层与内容层的有效分离,为用户提供了更加友好和动态的网页体验。这种技术的应用不仅提高了用户与网页的互动性,还为项目增添了更多创新和个性化的可能性。

在编码阶段,本项目采用了 MySQL 和 Django 作为数据库和服务器技术。MySQL 是一个开源的关系数据库管理系统,具有稳定和性能良好的特点。在开发过程中,我们可以针对数据库进行优化,例如索引和查询优化,以提高系统的查询效率和响应速度。Django 则是一个基于 Python 的开源 Web 框架,它提供了高效、简便的方式来构建 Web 应用。Django 采用了模型-视图-控制器(MVC)的架构,这有助于开发者组织和维护代码。此外,Django 拥有丰富的文档和活跃的社区支持,为开发人员提供了强大的工具和资源。

开发工具主要有 PyCharm、HttpReaquester 和网页解析器。PyCharm 作为一款专业的 Python 集成开发环境,提供了强大的代码编辑、调试和版本控制功能。它对 Django 框架的全面支持能够提高开发效率,同时还具备智能代码提示和自动完成的功能,使得编码更加便捷。HttpReaquester 是一个用于发送 HTTP 请求的工具,它在开发和测试阶段能够快速验证 API 的可用性。通过模拟不同类型的请求(GET、POST 等),可以有效地测试系统对外部请求的响应情况。在处理网页内容时,可能会用到一些网页解析器,例如 BeautifulSoup 或 LXML。这些工具有助于从 HTML 或 XML 文档中提取数据,用于系统对网页内

容的解析和处理。

画图工具使用的是 Microsoft 提供的一款流程图和示意图绘制工具 Visio。在系统开发中,使用 Visio 可以绘制系统架构图、流程图、数据库模型等,以便更清晰地传达系统设计和结构。

5.3.2 功能介绍

随着各种网站的成长,网络上的资源也是千奇百怪,当用户需要查询所需要的资源时,往往会被网络上的各种分享链接搞得晕头转向。想要从大量网站中找到最好、最新的资源也非常困难。所以提供一个高质量的网盘资源搜索平台无疑会为寻找资源带来更好便利。

本案例的资源搜索网站的功能主要分为两大模块,分别是前台模块和后台模块。功能架构如图 5-5 所示。

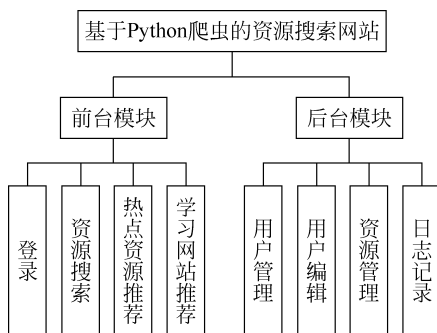


图 5-5 功能架构

网盘资源搜索网站的前台模块主要负责用户的搜索、热点资源的查看和搜索结果的返回,用户可单击搜索结果获取资源链接。主要功能如下。

(1) 登录。

用户可以通过两种方式登录本系统,具体功能如下。

① 验证码登录: 用户能够通过邮箱和验证码登录。

② 密码登录: 用户能够通过用户名和密码进行登录。

(2) 资源搜索。

搜索功能是本系统的核心功能,其功能主要是根据用户所输入的关键词获取相关度高的相关网盘资源,以改善用户的资源搜索效率并提高用户满意度。

用户在搜索栏输入关键词,单击“搜索”按钮,系统会在数据库中匹配关键词,当检索到

相关内容时则从数据库返回结果。若在数据库中没有含该关键字的内容就在线爬取。最后将结果返回给用户。搜索有全局搜索和分类搜索两种,具体描述如下。

① 全局搜索:覆盖整个网盘的资源库。为用户提供一个全面、高效的搜索工具,让用户能够方便地找到他们所需的网盘资源,而不受特定类型或分类的限制。

② 分类搜索:分为文档、音乐、视频、其他、图片类可选,选中会在其上显示打钩标志,更为迅速地得到用户所需要的网盘资源类型。

(3) 热点资源推荐。

热点资源功能是本系统的特色之一。该功能旨在向用户展示近期备受关注的资源标签,以便他们能够迅速了解当前的热点内容,从而提升用户在使用本系统时的体验感。通过这一功能,用户不仅能够快速了解当前备受瞩目的主题,还能根据个人兴趣深入挖掘更多相关资源。这样的设计使得系统更加用户友好,满足了用户对多样化资源的需求。

界面上会展示近期点击量较大的资源标签。用户可以根据个人兴趣点击其中的某个标签。选中标签后,系统将自动填充搜索框并开始搜索,并返回相关搜索结果。

(4) 学习网站推荐。

通过这一功能,用户可以获得一系列与其兴趣和技能相关的学习资源,为其提供更广泛的学习体验。这项功能主要通过展示一系列相关的技术标签来实现。这些标签旨在涵盖多个学习领域,包括但不限于 PHP、个人博客、前端开发、JavaScript、Hadoop 全书等。用户可以根据自己的学习兴趣和需求,单击这些网站推荐标签,轻松跳转到相应的学习资源网站。

这一特色功能的优势在于为用户提供了一站式的学习资源导航,节省了他们寻找相关资料的时间。无论是对于初学者还是专业人士,这个学习网站推荐功能都为他们提供了一个全面、可定制的学习平台。用户能够更深入地探索自己感兴趣的领域,拓展知识面,提高技能水平。

后台模块的主要对象是管理员,负责管理资源爬取、对数据库进行增删改和对后台用户的管理。主要功能如下。

(1) 用户管理。

用户管理功能是系统后台的核心之一,其主要任务是有效地管理用户信息、权限和安全。主要包括用户列表和用户搜索。系统后端提供一个清晰的用户列表,展示所有注册用户的基本信息。这包括用户名、角色、注册日期等关键信息。管理员可以通过用户列表快速了解系统中的用户情况,轻松进行用户信息的查看和管理。用户搜索方便管理员快速定位特定用户或用户组。搜索功能可以基于不同的标准,例如用户名、角色、注册日期等,从

而提供高效的用户管理体验。

这两个功能协同工作,使得管理员能够迅速而准确地管理用户信息。用户列表提供了整体视图,而搜索功能则允许管理员在庞大的用户数据库中快速找到特定用户。这样的设计旨在提高用户管理的效率和精确性,确保系统管理员能够轻松管理用户信息和权限。

(2) 用户编辑。

用户编辑功能的目的是为管理员或授权用户提供直观且易于操作的界面,以便他们能够方便地查看、修改和管理系统中的用户信息。通过提供这些功能,系统可以更灵活地适应用户管理的需求,确保用户信息的准确性和及时性。具体功能如下。

① 用户列表:显示系统中的所有用户,并为每个用户生成链接。通过点击链接,管理员或授权用户可以进入用户编辑界面进行相应的操作。

② 修改用户:在用户编辑界面,显示用户的基本信息,包括姓名、学号、兴趣爱好、创建时间等。管理员或授权用户可以对这些信息进行修改。提供删除、增加、保存等按钮,以便根据需求进行相应的操作。

③ 增加用户:提供一个界面,允许管理员或授权用户增加新的用户。在此界面,可以输入用户的姓名、学号、兴趣爱好、创建时间等信息,并进行修改。同样,也提供删除、增加、保存等按钮,以灵活管理用户信息。

(3) 资源管理。

资源管理功能的整体目标是为后台人员提供一个直观而强大的工具,以管理系统中的各类资源。通过这一功能,资源的操作、查看和搜索都变得更加方便,确保了资源信息的完整性和易用性。具体功能如下。

① 资源操作:后台人员通过资源操作功能可以在数据库中进行删改查的操作,包括批量删除数据。插入新记录时,输入网盘 URL、搜索关键字、资源类别、资源大小、创建时间、Reserve 字段,并可选择删除、增加、保存等按钮,提供了灵活的管理选项。

② 资源表格:显示资源的搜索关键字、资源类别、创建时间,并支持单字段排序。用户可以通过复选框选中资源进行批量删除,同时显示总资源条数并提供翻页和跳页功能,以确保系统中的资源信息一目了然。

③ 导航栏:提供资源的翻页、资源总数统计以及搜索和过滤搜索的功能。这样的导航栏设计使得用户能够快速定位和浏览系统中的各类资源,提高了资源管理的效率和便捷性。

(4) 日志记录。

日志记录功能旨在提供对系统操作的全面追踪和监控,帮助管理员了解系统的运行状

况,及时发现问题并采取适当的措施。通过这一功能,系统的安全性和稳定性得以增强。

管理员可以轻松排序显示并单击查看每一次操作的详细信息,包括操作类型、执行者和时间。这一功能不仅使管理员能够及时发现潜在问题,还提供了删除记录和撤销操作的灵活性,以确保数据库的完整性和系统的稳定性。在查看日志后,管理员可以直接返回到相应的操作界面,便于进一步处理或纠正。

5.3.3 数据库设计

本项目依照实体关系图,设计了 auth_user、auth_group 等表,如表 5-1 所示。这些具体设计如表 5-2~表 5-9 所示。

表 5-1 数据库表

缩写/术语	解 释	缩写/术语	解 释
auth_user	后台管理员	auth_group_permissions	操作权限分组
auth_user_groups	后台管理员分组	auth_user_user_permissions	管理员权限
auth_permission	操作权限	django_admin_log	管理员操作日志
auth_group	分组信息	person_wpurlset	爬取资源集

表 5-2 auth_user 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
id	int	11	用户 ID	NOT NULL PRIMARY KEY
password	varchar	128	用户密码	NOT NULL
last_login	datetiem	6	上次登录时间	NOT NULL
is_superuser	tinyint	1	是否为超级用户	NOT NULL
username	varchar	30	用户名	NOT NULL
first_name	varchar	30	名字	NOT NULL
last_name	varchar	30	姓氏	NOT NULL
email	varchar	254	邮箱	NOT NULL
is_staff	tinyint	1	是否为管理员	NOT NULL
is_active	tinyint	1	是否活动	NOT NULL
date_joined	datetime	6	加入时间	NOT NULL

表 5-3 auth_user_groups 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
id	int	11	组 ID	NOT NULL PRIMARY KEY
user_id	int	11	用户 ID	NOT NULL FOREIGN KEY
group_id	int	11	分组 ID	FOREIGN KEY

表 5-4 auth_group 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
id	int	11	分组 ID	NOT NULL PRIMARY KEY
name	varchar	80	分组名称	NOT NULL

表 5-5 auth_permission 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
id	int	11	权限 ID	NOT NULL PRIMARY KEY
name	varchar	255	权限名称	NOT NULL
content_type_id	int	11	内容类型 ID	NOT NULL
codename	varchar	100	操作名称	NOT NULL

表 5-6 auth_group_permissions 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
id	int	11	权限组 ID	NOT NULL PRIMARY KEY
group_id	int	11	分组 ID	NOT NULL FOREIGN KEY
permission_id	int	11	权限 ID	NOT NULL FOREIGN KEY

表 5-7 auth_user_user_permissions 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
id	int	11	用户权限 ID	NOT NULL PRIMARY KEY
user_id	int	11	用户 ID	NOT NULL FOREIGN KEY
permission_id	int	11	权限 ID	FOREIGN KEY

表 5-8 django_admin_log 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
id	int	11	日志 ID	NOT NULL PRIMARY KEY
action_time	datetime	6	操作时间	NOT NULL
object_id	longtext	0	操作对象 ID	
object_repr	varchar	200	操作对象	NOT NULL
action_flag	smallint	5	操作标志	NOT NULL
change_message	longtext	0	改变信息	NOT NULL
content_type_id	int	11	内容类型 ID	
user_id	int	11	用户 ID	NOT NULL FOREIGN KEY

表 5-9 person_wpurlset 表

数 据 项	类 型	长 度	取 值 含 义	约 束 条 件
wpid	int	11	网盘 ID	NOT NULL PRIMARY KEY
wpurl	varchar	100	网盘地址	NOT NULL
title	varchar	200	标题	NOT NULL
category	varchar	20	分类	NOT NULL
resourcesize	varchar	20	资源大小	NOT NULL
indextime	datetime	6	上传时间	NOT NULL
reserve	varchar	100	保留字段	NOT NULL

5.3.4 设计模式

1. 抽象工厂

场景：对于抓取到的数据，系统需要将它保存到数据库，有时也需要查询数据库。

抽象工厂模式类图如图 5-6 所示。工厂模式旨在提供一种创建对象的接口，将对象的实际创建延迟到具体的子类工厂中。客户端代码与实际创建的对象解耦，从而使系统更具灵活性和可维护性。对于给定的输入数据，定义两个不同的工厂意味着存在两个独立的工厂类，每个工厂类负责创建一组相关的对象。工厂类还包含了对对象的操作方法。这些操作方法可以根据业务需求执行各种功能，比如对数据库进行增、删、改、查等操作、合并两条数据库记录等。这两个工厂可以实现相同的接口或继承相同的基类，但它们的实现方式是不同的。

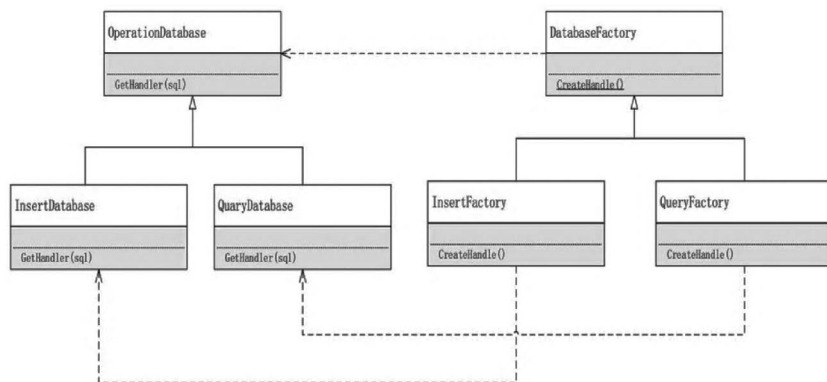


图 5-6 工厂模式类图

2. 装饰器

场景：在丰富数据库表时，系统需要读取 HTML 网页内容，有两种各有优缺点的获取方式。使用 Session，可以解决 ConnectionError 异常，但在一般情况下，使用 urlopen 效率会更高。因此我们需要根据不同的时机使用不同的方法，能达到更好的效果。

策略模式类图如图 5-7 所示。策略模式是一种行为设计模式，它定义了算法族，分别封装起来，使它们之间可以相互替换，从而使得客户端代码不受算法变化的影响。对于使用不同的方式读取网页内容的场景，策略模式可以通过定义不同的读取算法，比如从网络获取、从本地文件读取等，将这些算法封装到不同的策略类中。然后在需要读取网页内容的地方，通过使用合适的策略类来实现读取，而无须修改原有代码。这样就实现了读取方式的灵活性和可扩展性，而且符合开闭原则。

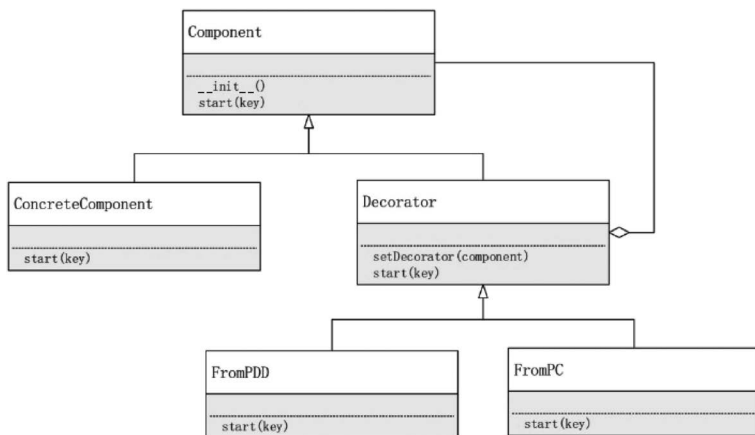


图 5-7 策略模式类图

3. 代理模式

场景：对于前端用户的输入，我们需要知道复选框的选择情况，以查询用户需要的信息，比如用户选择了视频，则后台返回视频分类的查询结果。

代理模式类图如图 5-8 所示。代理模式是一种结构型设计模式，其主要目的是通过引入一个代理对象来控制对另一个对象的访问。代理对象充当了客户端与真实对象之间的中介，通过代理可以在访问真实对象之前或之后执行一些附加的操作，比如控制访问权限、实现懒加载等。代理模式中通常有三个角色：抽象主题、真实主题、代理主题。通过这三个角色各自发挥作用可以实现对 categoryList 信息填充的灵活控制，同时将与信息填充相关

的逻辑从客户端代码中分离出来,提高系统的可维护性和可扩展性。

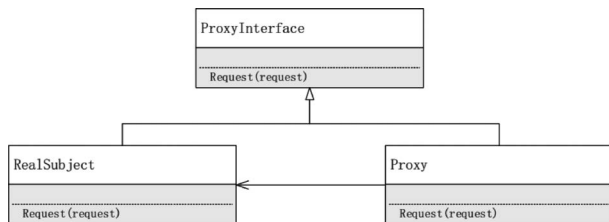


图 5-8 代理模式类图

4. 模板模式

场景：对于得到的 HTML 内容,需要使用解析方案,进一步得到网盘信息的具体内容,比如网盘地址、网盘资源大小等,具体分两层进行获取信息。

模块模式类图如图 5-9 所示。模块模式并非一种特定的设计模式,而更类似于一种组织代码的模块化编程方法。在模块模式中,代码被组织成一系列独立的模块,每个模块负责实现一个特定的功能,同时提供一个清晰的接口供其他模块使用。它可以使用 soup (soup 是 Python 中的一种 Dom 解析模块的一个方法),用来解析两层的网页内容,比如获取标题、时间。

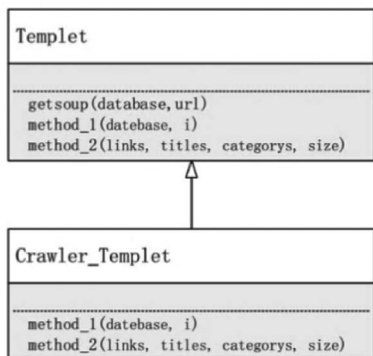


图 5-9 模块模式类图

5. 策略模式

场景：不同的网页可能有不同的解析算法,比如一种算法用于解析包含标题的网页,另一种算法用于解析包含时间的网页。

策略模式类图如图 5-10 所示。策略模式(Strategy Pattern)是一种行为型设计模式,它

定义了一系列算法,将每个算法封装起来,并且使它们可以相互替换,使得算法的变化不会影响到使用算法的客户端。策略模式主要关注的是定义一组算法、封装每个算法,并使它们可互换。策略模式通常包含三个角色:策略接口、具体策略类、环境类。通过使用策略模式,可以将这些解析算法封装成独立的策略类,使得客户端能够根据需要动态地切换使用不同的解析算法。

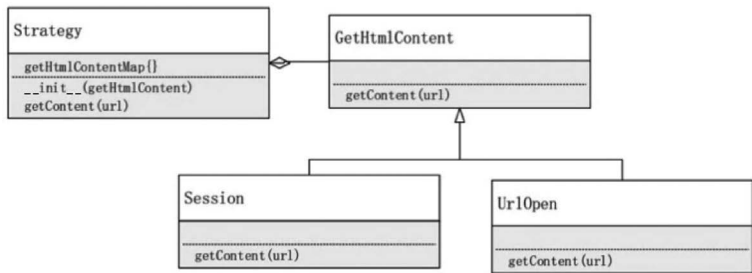


图 5-10 策略模式类图

5.3.5 出错处理措施

网络故障：客户端重新发送请求。网络故障对于性能的影响比较小,用户只要多一次的请求就可以解决。

服务器宕机：重启服务器。在发现服务器宕机之前,用户是无法访问系统的,此时对于用户的影响是较大的,改进方法是使用两台服务器,当其中一台发生宕机时,立即自动切换到另一台的服务器。

超时处理：在客户端与服务器之间的通信中,设置适当的超时时间。如果服务器在规定时间内没有响应,客户端可以认为发生了网络故障,触发重新发送请求或其他相应的处理。

备份和恢复：定期对系统数据进行备份,以便在发生灾难性故障时能够快速恢复。备份可以包括数据库、配置文件等关键数据。

5.3.6 测试分析

本节旨在总结测试阶段的过程和分析结果,以确定系统是否满足需求。测试分析报告记录和分析测试结果,是测试过程中的重要环节,对软件性能进行综合评估,指出不足之

处。这有助于未来改进软件功能,提高性能,并帮助开发者理解源代码,优化程序。完成测试后,需提交测试计划执行情况说明,对结果进行分析,并提出结论和建议。

以下将从后台测试、前台测试和爬虫测试几部分进行总结分析。

(1) 后台测试。

后台测试总结如表 5-10 所示。

表 5-10 后台测试总结

模 块	测 试 功 能	测 试 结 果
用户表格	显示总共用户数、电子邮件地址、姓名、职员状态、任意字段排序	返回用户列表
	已知用户的用户名	返回用户列表
导航栏	进行用户搜索、过滤搜索	返回新的用户列表
	按钮可增加用户	返回新的用户列表
用户列表	显示用户的所有人/形成链接/进入用户编辑界面	返回用户个体列表
修改用户	显示用户的姓名、学号、兴趣爱好、创建时间	返回用户个体信息
	选择删除、保存并继续编辑	删除、保存并继续编辑
	保存并增加一个	保存并增加一个
	保存	返回用户个体信息
增加用户	增加用户	返回新用户
	输入用户的姓名、学号、兴趣爱好、创建时间、修改	返回新用户
	删除	删除
	保存	保存
	继续编辑	继续编辑
	保存并增加一个	保存并增加一个
增加资源	插入一条记录	返回新的资源记录
	继续编辑	继续编辑
	保存并增加一个	保存并增加一个
	保存	返回新用户
资源表格	显示资源的搜索关键字	返回操作后的资源表格
	复选框中选中	选中
	批量删除	删除
	显示总共的资源条数	显示
	翻页跳页功能	翻页跳页
导航栏	资源的翻页	返回操作后的资源表格
	过滤搜索	返回操作后的资源表格
我的动作	记录管理员最近对数据库进行的所有操作	记录
	排序显示	显示
	可进行单击,回到操作界面	跳转
模块	测试功能	测试结果
用户表格	显示总共用户数	返回用户列表

根据表 5-10 可以发现,测试情况总体上看是比较良好的。用户表格、导航栏、用户列表、修改用户、增加用户、增加资源以及资源表格等各模块的功能测试都达到了预期的效果。例如,用户表格能够正确显示用户信息并支持排序功能,导航栏能够进行用户搜索和过滤搜索等操作,用户列表能够显示用户的个体信息并形成链接,修改用户和增加用户等功能也都能正常执行,资源表格能够显示资源的搜索关键字、类别和创建时间等信息,并支持翻页跳页功能。此外,针对我的动作模块,记录管理员对数据库的操作、排序显示以及单击跳转等功能也得到了验证。

总的来说,后台测试显示了系统在各个功能模块上的稳定性和可靠性,为系统的后续部署和应用提供了坚实的基础。

(2) 前台测试。

前台测试总结如表 5-11 所示。

表 5-11 前台测试总结

模 块	测 试 功 能	测 试 结 果
欣赏美图	由最新的美图中选择四张,这四张图片会不断地滚动播放	欣赏
极光团队	介绍团队名	显示
	单击进入会显示一个动画	极光,动画效果
搜索框	显示“请输入关键词”	显示
	在其中输入用户想要搜索的网盘资源	保存结果,等待点击搜索键提交
分类选项	文档、音乐、视频、其他、图片	打钩,成功勾选
	多选	多选成功,多个字段返回结果
热门推荐	单击编程、游戏、美国队长、青春、剧场版、OVA	包含查询的热门推荐的新动态页面
搜索键	单击可对输入关键词进行搜索	包含查询到的信息的新动态页面
站内通知	显示管理员想说的事件与公告,可留言	暂未实现,链接到对应的贴吧
标签云	用户可点击进入相关网址学习计算机技术	计算机学习网站
微信关注我	用户扫码	可加好友或者收藏网站
友情链接	链接到对应的友情网站	友情网站首页
统计导航栏	统计此次搜索总共有多少条数	此次搜索结果共#条
标题	跳转到百度网盘相关资源首页	百度网盘页面
分类标签	显示资源类型	显示资源的 category 字段
上传时间	显示资源上传时间	显示资源的 time 字段
评论	显示资源评论	暂未实现
浏览	显示资源浏览数	暂未实现
资源简介	显示资源简介	计划实现百度爬取
底部	资源页面查询结果的翻页	可翻页
	跳转页面浏览	跳转页面成功
	首页和尾页浏览	链接到相关的页面,显示具体信息

前台测试表明,系统在大部分功能模块上表现良好,但也有一些暂未实现或待改进的地方。首先,就已实现的功能而言,欣赏美图、极光团队介绍、搜索框、分类选项、热门推荐、搜索键、标签云、微信关注我、友情链接、统计导航栏、标题、分类标签、上传时间和底部翻页等功能都得到了验证,能够顺利执行对应的操作或显示所需信息。然而,也存在一些功能暂未实现或需要进一步改进的地方,如站内通知、评论、浏览、资源简介等功能。这些功能的实现或改进将有助于提升用户体验,增加系统的吸引力和实用性。

综上所述,尽管系统已经实现了大部分功能并且表现良好,但还需要对一些功能进行完善和优化,以确保系统能够满足用户的需求并提供良好的使用体验。

(3) 爬虫测试。

对于爬虫部分,本项目也进行了一系列的测试,总的来说,虽然经历了一些挑战和困难,比如网络传输的保密性和安全性、数据库的安全性还存在一定的缺陷,有可能会由于传输过程中数据的丢失造成软件运行的错误,同时对于限定性输入框中的限定条件不够完整。但在测试过程中积累了丰富的经验,对爬虫的效率和速度进行了测试,并成功解决了一些翻页和请求失败的问题,为后续的爬虫工作提供了宝贵的经验和教训!

5.4 小结

在本项目中,搜索引擎采用自有搜索技术,通过自主开发的爬虫程序,从网络上抓取共享文件信息,建立自己的资源数据库。相比国内外其他的网盘搜索引擎,该系统旨在提供更高相关度的搜索结果,并且界面简洁、操作快捷,为用户提供优质的搜索体验。主要研究内容包括爬虫的开发、网站的搭建、数据库的构建以及框架的应用。爬虫程序是项目的核心,负责从网络上收集资源信息;网站的搭建涉及前后端页面的设计与管理;数据库用于存储从网络上获取的资源信息;框架的学习应用旨在提高开发效率。

总的来说,通过采用自有搜索技术、Python 开发语言、MySQL 数据库和 Django 框架,我们期待看到一个更为高效、准确的网盘搜索引擎为用户带来更便捷、优质的资源检索体验,为互联网用户提供更全面、高效的资源共享平台。本案例应用于福州大学计算机与大数据学院电子信息专业学位硕士研究生课程“高级软件工程”“软件体系结构”等,累计授课超过 500 人次,取得了优异的教学效果。