

基于Spring Cloud框架的档案信息资源共享利用服务系统

王腾军¹, 马强¹, 李京林², 戴玉刚¹, 徐涛^{1*}

¹西北民族大学中国民族语言文字信息技术重点实验室, 甘肃 兰州

²甘肃省档案馆, 甘肃 兰州

收稿日期: 2022年6月20日; 录用日期: 2022年7月19日; 发布日期: 2022年7月27日

摘要

本文以档案信息资源管理为基础, 在分析传统档案管理工作流程的基础上, 分析设计并实现了资讯管理、政策法规、档案收集、查档管理、档案馆管理、角色管理、用户管理、档案统计及系统管理于一体的档案信息资源共享服务系统, 该系统可以提高档案管理效率, 为开展数字化档案管理工作提供信息化支持。本系统采用前端与后端分离的开发模式, 分别采用Vue框架与Spring Boot + Spring Cloud框架进行开发, 经过性能测试, 该系统在功能和可靠性上都达到了预定目标。另外, 本系统不仅满足档案使用者查阅档案的需求, 而且全过程收集的数据可以为档案领域科研工作提供强有力的支持。

关键词

档案信息资源管理, Vue框架, Spring Cloud框架

File Information Resource Sharing and Utilization Service System Based on Spring Cloud Framework

Tengjun Wang¹, Qiang Ma¹, Jinglin Li², Yugang Dai¹, Tao Xu^{1*}

¹Key Laboratory of Information Technology for Chinese Ethnic Languages and Scripts, Northwest University for Nationalities, Lanzhou Gansu

²Gansu Provincial Archives, Lanzhou Gansu

Received: Jun. 20th, 2022; accepted: Jul. 19th, 2022; published: Jul. 27th, 2022

*通讯作者。

文章引用: 王腾军, 马强, 李京林, 戴玉刚, 徐涛. 基于 Spring Cloud 框架的档案信息资源共享利用服务系统[J]. 计算机科学与应用, 2022, 12(7): 1839-1854. DOI: 10.12677/csa.2022.127184

Abstract

Based on the management of file information resources and the analysis of the workflow of traditional archives management, this paper analyzes, designs and implements an archives information resource sharing service system integrating information management, policies and regulations, archives collection, file retrieval management, archives management, role management, user management, archives statistics and system management. This system can improve the efficiency of archives management, provide information support for the development of digital archives management. The system adopts the development mode of separating the front end from the back end, and uses Vue framework and Spring Boot + Spring Cloud framework to develop. After performance test, the system has achieved the predetermined goal in terms of function and reliability. In addition, the system not only meets the needs of file users to access files, but also the data collected in the whole process can provide strong support for scientific research in the field of archives.

Keywords

File Information Resource Management, Vue Framework, Spring Cloud Framework

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

在信息化快速发展的背景下，档案领域的信息量规模进一步扩大，给档案管理部门带来了巨大的挑战。现有的各级档案资源共享服务系统主要以馆藏介绍、目录查询等为主，缺乏现代网络技术进行传递和交流，而且信息资源受空间、数量等限制比较分散，缺乏标准化信息资源共享体系和系统开发应用，多次开发已经使得档案信息管理系统的业务逻辑出现混乱、功能重复、数据安全性差等问题，不能满足广大档案利用者的多样化需求。基于以上背景，需要档案信息管理系统在保障系统稳定运行的情况下，又可以根据特殊的档案信息管理工作场景需求进行单独的扩展，方便档案使用者查阅档案信息，从而提高档案管理者全过程工作效率，成为档案行业急需解决的问题。

本文以甘肃省档案局提供的档案信息资源为基础，以整合档案资源、促进区域档案管理部门信息社会化服务为目标，运用先进的信息网络技术，打破地域限制，为广大档案利用者打造一个统一的、一站式的档案信息资源共享和服务平台，实现区域性档案信息资源的共享利用，为不同用户提供全方位、多角度、深层次的档案信息服务。通过该系统的研究与实现，将全面整合区域各级开放档案信息资源、各类专题档案、各种民生档案等档案资源，为区域档案用户提供便捷化档案利用服务。

本文采用前端与后端分离的开发模式，以 Vue 框架与 Spring Boot + Spring Cloud 框架为基础，构建与实现档案信息资源共享服务系统，该系统可以对档案数据进行全过程采集和管理，从而实现对档案的全过程管理目标。经过测试和试用，该系统功能全面及性能稳定，可以减轻档案管理人员的工作负担，提升档案管理效率，为管理人员、档案使用者提供便捷化和信息化服务。

2. 微服务架构

2.1. 传统的单体构架

单体架构的档案共享服务系统因为开发时间较早，当时的应用场景比较单一，系统提供的功能较少，

对业务场景没有划分，所有的业务模块都编写在一个项目中，但随着时间的推移，数据量不断增加，业务逻辑关系复杂、代码可读性差、升级维护工作难度大等问题逐渐显露[1]。图 1 为常用单体应用架构图。

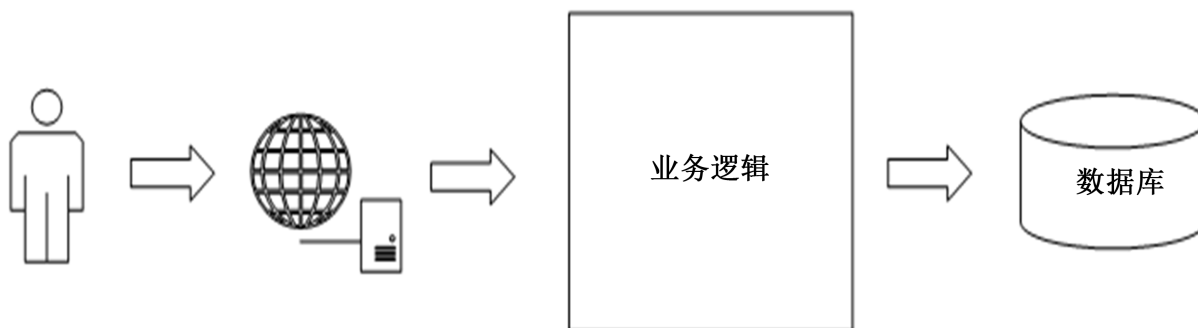


Figure 1. Architecture diagram of single application

图 1. 单体应用架构图

当前，档案信息化系统需求功能随着应用场景的多元化在不断更新、持续增加，传统的单体架构的档案信息管理系统无法满足新形势下档案业务快速变化的需要，容易产生各种各样的问题，主要表现在以下几个方面。

1) 过度的复杂性。档案信息化的方向不断发展，整个档案共享服务系统包含的功能模块越来越多，一些功能模块之间的逻辑关系和依赖关系不清晰，功能模块的边界模糊，将会使得整个系统变得异常复杂，会对整个档案信息管理工作带来不可逆的严重影响。

2) 开发速度慢。档案信息管理系统的需求不断变更，再加上系统开发人员及管理部门人员的更迭，会逐渐积累应用程序的技术债务。初期设计的功能或者代码难以修改，系统开发人员难以根据现有档案信息管理工作要求选择适合的技术完善系统功能。

3) 部署周期长。单体架构的档案信息管理系统应用中，随着系统功能的更迭，项目的代码量也在不断的增加，从而导致系统构建和部署的时间也随之增加，再加上每次功能的变更或 BUG 的修复，都需要重新部署整个应用，这使得应用项目上线部署的效率大幅降低。

4) 系统难以扩展。系统不能做到按需扩展，只能作为一个整体进行扩展，无法很好的适应当前档案信息管理工作业务的特点，尤其在档案信息管理这类专业化和复杂化的系统中，因为扩展能力有限，整个项目的各个模块变得很不灵活。

2.2. 微服务架构

微服务架构是针对传统单体架构复杂度高、不易扩展等缺点出现的一种新的架构概念。它是一种架构模式，将单一应用程序划分成多个小的服务，多个小服务之间彼此协调、互相配合，为用户提供完整功能。微服务架构风格是将每个小型服务运行在自己的进程中，这些服务围绕具体业务进行构建，服务间通信采用轻量级通信机制，通过全自动部署机制独立部署，每个服务可根据具体的业务需求，用不同的语言开发，数据存储技术也可以多样化[1]。

该架构应用到档案信息管理系统中，整个系统中的每一个服务运行在独立的进程中，服务与服务之间采用轻量级的通信机制进行沟通。每个服务均依据具体业务进行划分，对应的每个服务都能够被独立安放到档案管理工作生产环境中。此外，为了尽量避免原系统中高耦合、低效率的服务管理机制问题，对具体的每个服务而言，可以根据具体档案信息管理工作上下文，选择不同的语言与数据库等技术对其进行构建。微服务架构模式从根本上解决了传统单体架构系统所面临的问题，为系统后期的维护升级留

下了足够的空间，系统在后续开发的过程中，开发人员只需要通过阅读原系统中的几个微服务，了解工作机制后，便可以进入开发流程，得益于微服务架构模式，开发人员对系统需求的技术选型，将能更好的适用于整个功能的开发，有利于提升系统整体的部署和使用效率。从而实现从系统服务等级、安全要求、档案信息管理业务监管等多个维度针对不同的服务实现不同的治理。

总体来看，微服务架构方式能够很好的将新型技术应用到档案信息管理工作中去，充分保障系统稳定运行，该架构模式应用于档案信息管理系统中，具有鲜明的优势。图2是常用的微服务架构模型图。

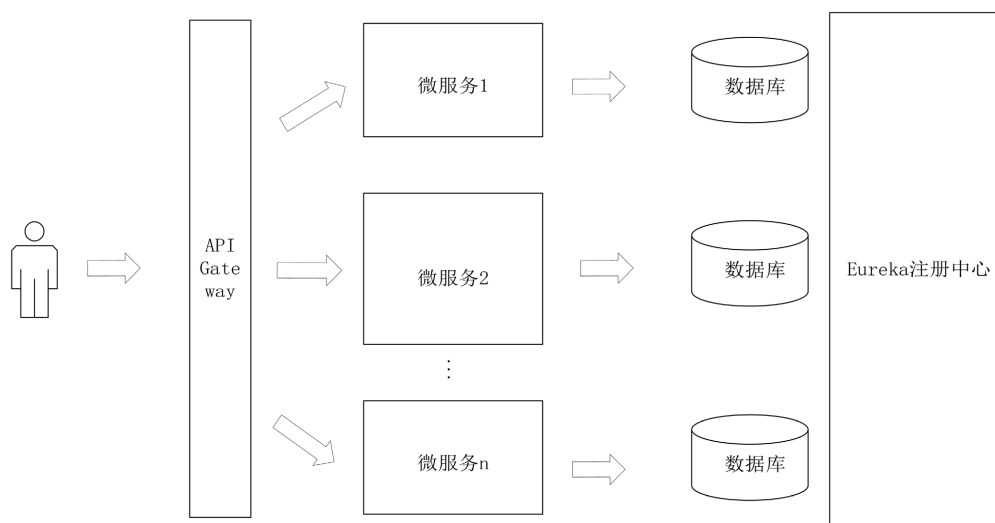


Figure 2. Microservice architecture diagram
图2. 微服务架构图

1) 微服务架构着力解决了传统单体应用复杂度高的困难，将传统单体应用分解为多个服务，开发过程中只需要着手解决每个服务的问题，有效降低了项目的复杂度。同时，在详细分析系统业务需求的前提下，对于采用传统单体架构很难实现的功能，微服务架构模式通过将应用分解为多个服务，每个服务都通过一个RPC或者消息驱动API定义清楚的范围，模块化解决方案为系统的开发提供了十分便利的条件。

2) 在技术层面上没有限制，采用微服务架构，对于每个微服务的开发，微服务之间松耦合，服务内部高内聚的特点，使得不同的开发团队可以来维护不同的微服务，整个开发工作会变得更为专业化，开发者可以根据具体考试工作业务需求，来选择最佳的开发技术和具体的实现方案，每个微服务在工作过程中，只需要对外提供API接口服务即可。这使得原始的项目能够适应技术更新发展，从而实现在技术上的扩展，而不会被限制在项目最初采用的技术。

3) 微服务架构中每个服务都是完全隔离的，所以一个服务造成内存泄漏只影响该服务，不会影响其他服务，其他服务正常处理请求，相较于传统单体架构应用一旦系统发生了故障，那么整个应用可能就没法使用的缺陷，微服务架构具有更好的故障隔离的特点，这也为新技术在系统中的试用和实验提供了便利条件，因为微服务体系结构中的每个服务都相对较小，所以自动化测试更容易编写，执行也更快。

3. 系统相关技术

本系统采用前端与后端分离的开发模式，后端开发语言主要采用Java，考虑到档案共享服务系统的模块多、业务逻辑复杂、后期扩展性要求较高等特点，在后端整体架构选择方面，本文创新性的采用了当前社区活跃度很高的Spring Cloud微服务架构，方便前期模块拆分以及后期功能扩展；针对档案共享

服务系统数据量大、并且结构化明显的特点，采用当前行业非常流行的基于结构化语言操作的关系型数据库 MySQL；前端框架采用构建用户界面的渐进式 Vue 框架。

3.1. Spring Cloud 微服务架构

Spring Cloud 是基于 Spring Boot 实现的微服务架构，它为微服务架构中涉及的服务管理、断路器、智能路由、配置管理、控制总线、分布式会话和集群状态管理等功能提供了一系列解决方案和具体实现方式[2]。档案共享服务系统功能全面复杂，Spring Cloud 架构能有效降低系统开发过程中的隐含问题，提升开发效率。图 3 是简单的 Spring Cloud 微服务架构模型图。

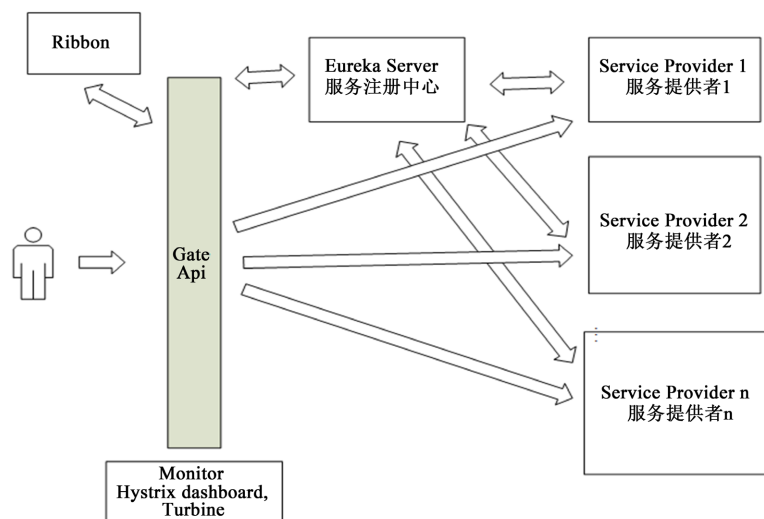


Figure 3. Spring Cloud microservice architecture diagram
图 3. Spring Cloud 微服务架构图

Spring Cloud 是当前开发大型 Web 应用最为流行的后端架构之一，采用微服务去中心化理念，可以将整个档案信息资源共享服务系统项目划分成多个独立的、自治的微服务。考虑到本文研究的档案信息资源共享服务系统具有数据量大，功能模块多、业务逻辑复杂等特点，因此采用微服务技术将复杂的业务逻辑按模块划分为一系列小的独立的 Web 服务，单独进行开发、部署和运行，可以减少各个模块之间不必要的耦合，同时各微服务通过统一的网关对外提供服务，又可以根据特殊的需求单独地进行扩展，为后期档案信息管理系统系统升级和功能扩展提供了便利。本文选用 Spring Cloud 微服务架构优势主要表现在以下几个方面。

- 1) Spring Cloud 诞生于 Spring 大家族，因此与生俱来的可以和 Spring 完美兼容。
- 2) Spring Cloud 社区活跃度很高，相关的书籍教程很丰富，遇到问题很容易找到方案解决。
- 3) Spring Cloud 的一整套微服务组件可以轻松帮助我们完成服务熔断、客服端均衡负责、注册中心等各种功能。

3.2. Spring Boot 框架

Spring Boot [3]是一个基于 Spring 项目开发全新框架，它使用了特定的方式来进行配置，简化了 Spring 应用的初始搭建以及开发过程。Spring Boot 的出现极大简化了 Spring 应用程序的开发，作为 Spring 的一套快速配置脚手架，可以使用 Spring Boot 快速搭建单个微服务，采用 Spring Boot 框架开发微服务具有以下几个特点：

1) 项目构建简单。Spring Boot 是伴随着 Spring 4.0 产生的,其作用就是帮助开发人员快速的搭建基于 Spring 框架的项目,因此 Spring Boot 继承了 Spring 的优点,这使得开发 Spring 项目更加方便快捷。

2) 避免了编写大量的样板代码。Spring 虽然是 Java EE 轻量级框架,但是它的配置文件很多,各种 Annotation、XML 配置很难维护,尤其是随着代码量增多,一些配置信息也越来越多。Spring Boot 是采用 Java 类配置的方式,而且 Spring Boot 项目的全局配置文件只需要一个 application.xml 就足够了,使开发人员摆脱基础的配置工作,能够更加的专注于业务逻辑的开发。

3) 简化部署及提高生产力。在部署普通的项目时,首先需要在服务器上安装 tomcat 中间件,然后把项目打成 war 包部署到 tomcat 容器中。但是使用了 Spring Boot 后,我们可以直接采用 Spring Boot 内嵌的 tomcat,我们只需要将项目打成 jar 包,使用 java-jar xxx.jar 一键式启动项目,降低对运行环境的基本要求,不需要额外的 web 容器,只需要环境变量中有 JDK 即可。

4) 简化监控。功能模块的部署是分布式的,大部分功能模块都是运行在不同的机器上,彼此通过服务调用进行交互,前后台的业务流会经过很多个微服务的处理和传递,出现了异常需要快速定位是哪个环节出现了问题,在这种框架下,微服务的监控显得尤为重要,通过引入 spring-boot-start-actuator 依赖,直接使用 actuator 提供的 REST 端口来获取进程的运行期相关性能参数来进行系统性能监控。

3.3. 数据库技术

MySQL 数据库是当前 IT 领域最流行、使用最为广泛的数据库管理系统,也是一个支持多线程、高并发的关系型数据库管理系统,是基于 Linux 操作系统开发的一种开放源代码的关系型数据库管理系统(RDBMS),使用结构化查询语言(SQL)进行数据库管理[4],SQL 主要用于对关系数据库系统中数据进行增删改查的一种非过程化的编程语言。开源是 MySQL 数据库的优点之一,除此之外,MySQL 数据库还有很多优势。比如:在数据库连接途径的选择方面,它提供 TCP/IP、ODBC 和 JDBC 等多种数据库连接途径,在此基础上,我们可以通过多种编程语言对数据库进行操作,常见的有 Java 和 C,可以方便的调用 MySQL 数据库,实现数据存取操作,同时在查询速度方面,MySQL 采用优化的 SQL 查询算法,对提高查询速度,能起到有效的促进作用;而且 MySQL 还具有支持多线程的优势,能做到对 CPU 资源的充分利用;MySQL 还提供各类管理工具,从而实现管理、检查、优化数据库操作的。综合上述情况来看,MySQL 数据库在速度、可靠性和适应性等方面都表现的极为出色,能够满足共享服务系统的日常运行需求[5]。

3.4. Vue 前端框架

Vue 前端框架是构建用户界面的渐进式轻量框架,Vue.js 是一个 JavaScriptMVVM 库[6]。它以数据驱动和组件化的思想构建,采用自底向上增量开发的设计模式,区别于其他重量级框架[7]。Vue 的核心库关注视图层,为开发者提供了简捷且易于理解的 API,具有易于学习的特征,很容易与其它库或已有项目进行整合,而且 Vue 自身并非是一个全能框架,学习门槛比较低。在与相关工具和支持库一起使用时,Vue 也能完美地驱动复杂的单页应用,同时驱动采用单文件组件和库开发的复杂单页应用。

4. 系统设计

4.1. 系统整体构架

该系统存在两个离散的数据域,分别是前台信息门户数据域以及后台档案管理数据域,意味着需

要使用多个微服务。结合系统功能性需求和微服务架构分析情况来看,将档案信息资源共享服务系统整体分为门户信息服务、档案服务、查档服务、基础服务和一个网关服务,主要采用 Spring Boot 框架和 Spring Cloud 架构搭建各微服务模块[8]。每个微服务架构层次分为控制层,业务层和持久层,如图 4 所示。

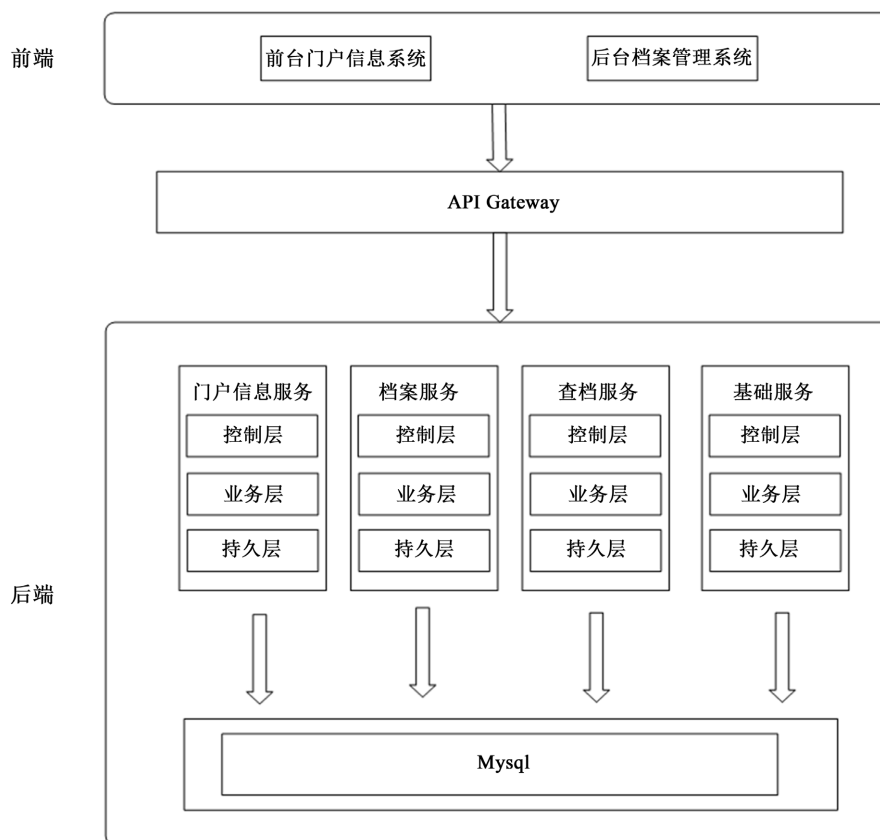


Figure 4. Overall architecture of archives information resource sharing service system
图 4. 档案信息资源共享服务系统整体架构图

控制层主要是对进来的请求进行数据校验以及封装,以保障请求数据的合规合法性;业务层主要针对本微服务的功能进行模块划分,实现 MySQL 数据库的增删改查等操作;持久层主要实现档案管理系统中的普通数据和档案信息的存储,主要由 MySQL 实现存储;系统使用微服务构建业务逻辑,接口采用 REST 理念进行构建,将 HTTP 作为服务调用协议,使用 URI 传达意图,请求和响应使用数据序列化协议 JSON-JavaScript。

4.2. 系统功能模块分析

基于 Spring Cloud 微服务的思想,通过对档案信息资源共享服务系统关键功能模块的拆分,本系统拆分出 8 个微服务功能,分别为“资讯管理”、“政策法规”、“档案收集”、“查档管理”、“档案馆管理”、“角色管理”、“用户管理”和“档案统计”。微服务利用 Spring Boot 框架迅速搭建,微服务注册信息通过 Spring Cloud 提供的一系列微服务组件进行统一管理与发现,较好的实现了档案管理部门档案信息化的管理,有效缓解了“信息孤岛”和档案信息共享问题。系统详细功能模块具体拆分如图 5 所示。

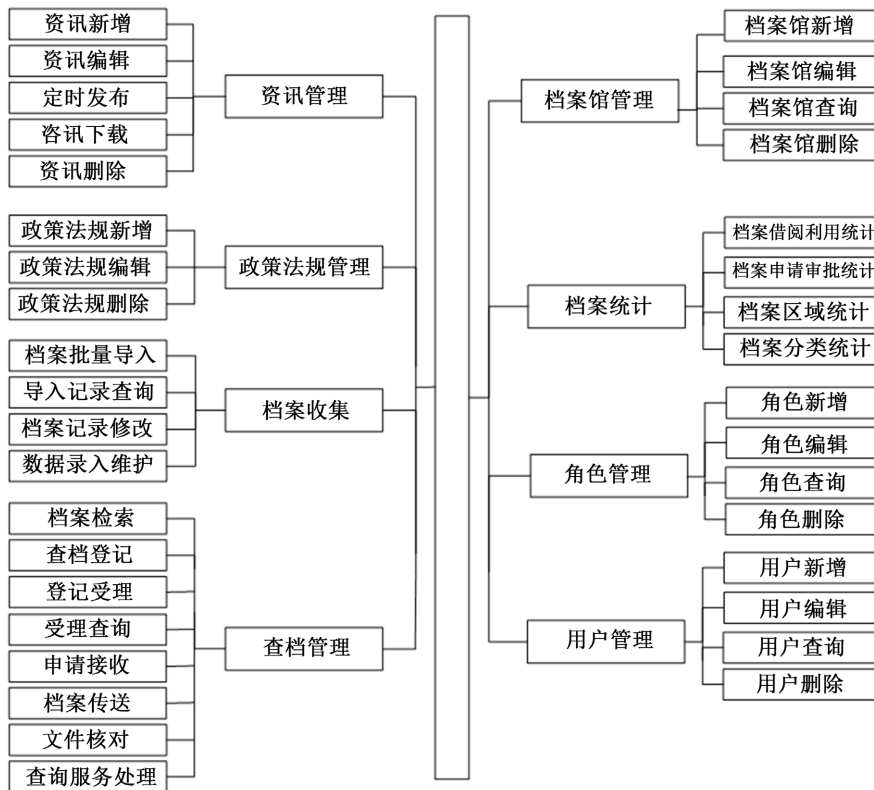


Figure 5. Detailed functional module division of the system
图 5. 系统详细功能模块划分

4.3. 系统详细设计

档案信息资源共享服务系统共包含八个功能模块，这里主要以资讯发布管理模块、档案收集模块、查档管理模块和档案统计模块为例进行分析。

4.3.1. 资讯发布管理模块

资讯发布管理是系统首要涉及的部分，也是门户信息子系统中重要的部分，平台通过发布的资讯信息向用户传递关于平台的相关信息，主要涉及发布的时间和载体。系统管理员用户在档案管理子系统中将对应的资讯信息编辑后发布出去，然后普通用户可以在门户信息网站查看到该资讯信息，保证用户即时了解到平台的资讯动向。

系统的资讯发布管理业务流程如下：首先后台管理者填写包含标题、发布时间以及资讯主体内容在内的资讯相关信息，完成填写后，点击确认添加按钮，若填写的信息存在错误或格式不规范的，则系统给出明确提示，并修改后重新提交保存；在资讯新增成功之后，在资讯列表页面选择刚添加的资讯，点击发布按钮，即可实现资讯发布。若系统发布正确，此时可以在前台门户信息系统的资讯页面显示发布的资讯信息。图 6 所示是资讯发布管理程序流程图。

4.3.2. 档案收集模块

系统的档案收集业务的流程：后台管理者进行数据导入页面，首先下载文件导入模板，按照模板要求，整理好需批量上传的档案信息后导入，系统验证导入文件是否符合文件模板要求，若验证未通过，则表示导入文件信息格式错误；若验证通过，则系统会解析导入文件的档案条目，分别存储各个档案信息，并提示导入成功以及显示批量导入的档案信息列表。下图 7 所示是档案收集模块程序流程图。

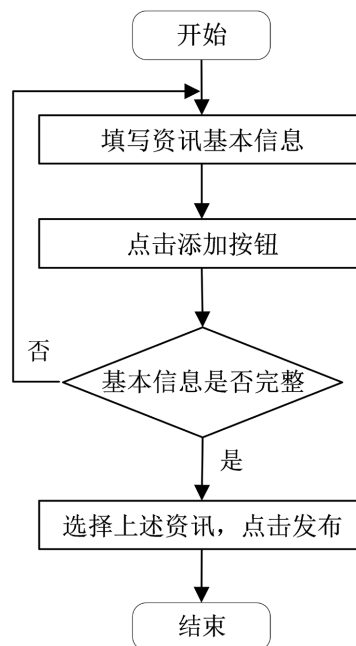


Figure 6. Flow chart of information release management procedure

图 6. 资讯发布管理程序流程图

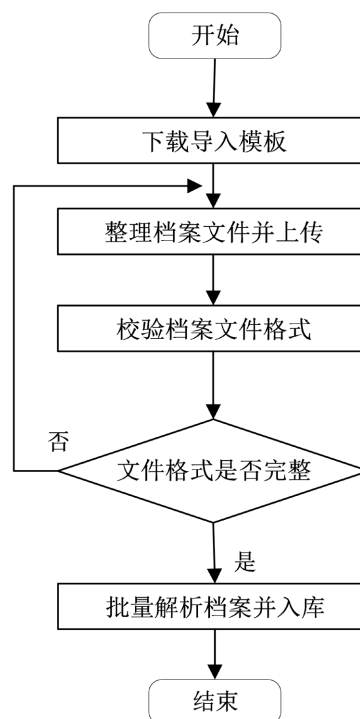


Figure 7. Flow chart of archives collection procedure

图 7. 档案收集程序流程图

4.3.3. 查档案管理模块

系统的查档管理流程：首先用户进入档案检索页面，通过筛选条件查询需要查阅的档案信息，选择指定档案条目，点击查档登记按钮，填写查档申请表单，然后点击提交按钮，验证数据格式是否正确，

若验证未通过，则系统给出错误提示，引导用户重新填写；若验证通过，则系统给出提交成功提示。

用户进入登记受理页面，选择指定的待受理的查档登记记录，审核借阅登记信息，若信息合格，则受理通过，查档申请进入申请接收环节；若受理不通过，则查档申请就此结束。图 8 是查档管理业务流程图。

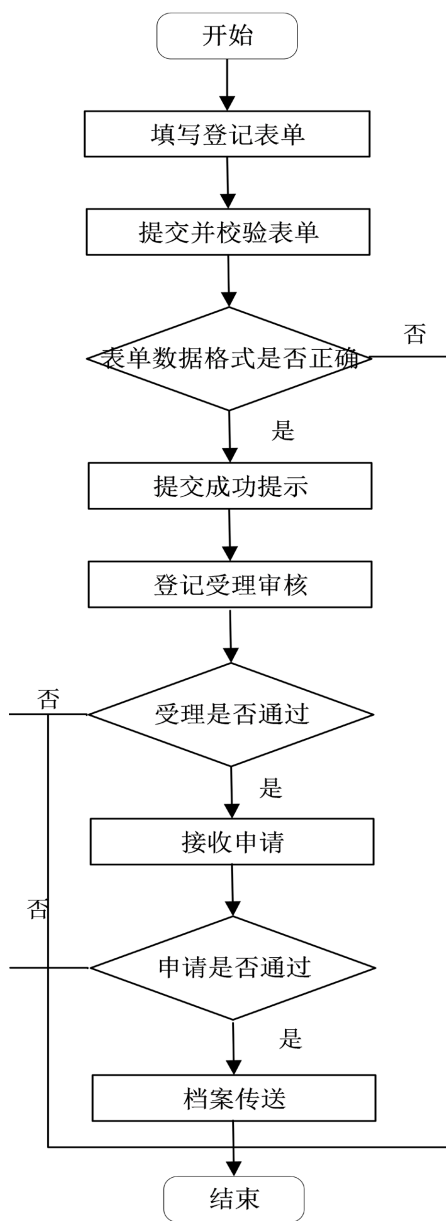


Figure 8. Process flow chart of file access
图 8. 档案查阅程序流程图

4.3.4. 档案统计模块

本系统使用的数据库为 MySQL，以各档案馆档案信息、用户查档行为等数据作为源数据，然后通过这些源数据建立相应的数据视图，继而在系统的数据统计及分析模块的相应页面上提供基于多条件的数据统计和汇总的查询操作。

该模块的主要功能如下：借阅利用统计，按年度统计各档案馆档案的借阅/利用次数柱状图和本馆近一年来每个月档案借阅/利用次数折线图；申请审批统计，按年度统计各档案馆档案的申请件数柱状图和本馆近一年来每个月档案借阅申请审批情况折线图；档案区域统计，实时统计甘肃省各个州市的档案数量分布情况，并通过地图直观可视化展示；档案分类统计，实时统计各档案馆中各类别档案的数量以及各自占比，并通过饼状图直观展示。

4.4. 数据库设计

根据系统整体运行及档案信息数据规范化建设起着支撑作用，设计了以下业务表：档案馆表 `tb_archive_holls` 记录档案馆信息；档案分类表 `tb_archive_cates` 记录档案分类信息；档案信息表 `tb_archives` 记录档案信息；档案批量导入表 `tb_archive_imports` 记录导入文件存放信息；档案查阅申请表 `tb_archive_apply` 记录档案查阅信息；档案查询申请处理流水表 `tb_archive_apply_flows` 记录档案查询申请处理信息[8]。

本系统涉及到的关键实体主要有一下几项：档案馆、角色、用户、档案导入记录、档案、档案分类、查档申请、档案处理流水、资讯、政策法规等。图 9 是系统中主要实体的 ER 图：

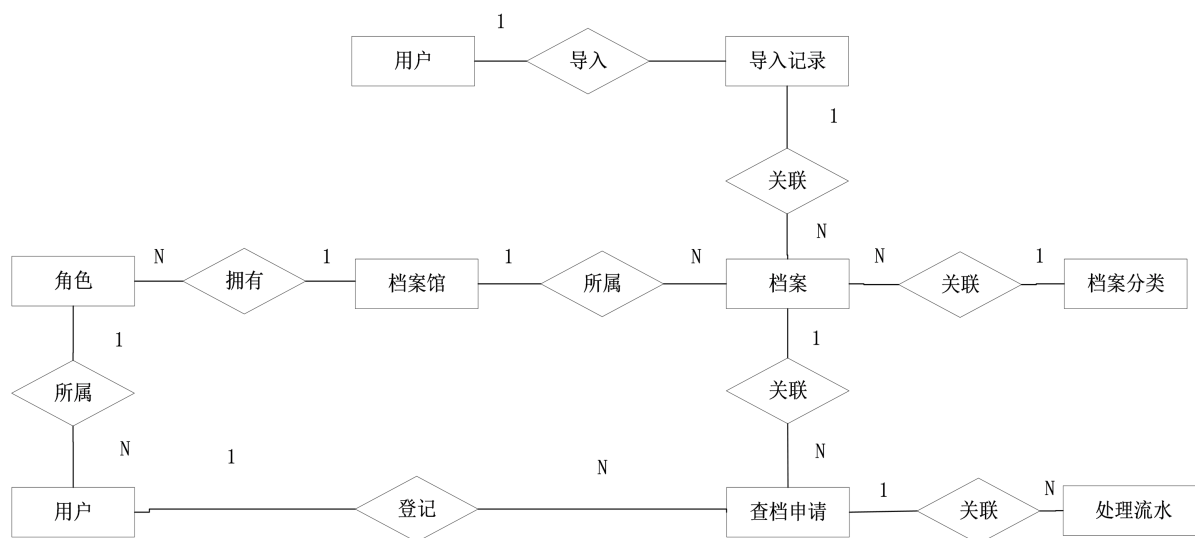


Figure 9. Overall ER diagram of the system

图 9. 系统总体 ER 图

4.5. 系统应用对比分析

随着档案信息化建设的发展，各级档案部门积累了大量数据，数字化档案成为现阶段发展重点，档案信息化系统逐渐取代人工收集与利用档案，而原有的信息化系统存在为用户提供单一检索、数据无法共享利用等问题。

本文档案信息资源共享利用服务系统在某档案馆自 2022 年 2 月份试运行以来，在对比原有档案信息化系统档案调用次数后，档案管理效率明显提高，原本需要 1 h 的档案管理任务量，现在仅需 45 min 左右即可完成，效率提高了 25%；在以往的档案信息化系统中，各级档案部门的数据共享利用基本为零，而本系统自上线运行后，信息共享利用接口调用次数明显提高，信息共享交换量达到了日均 1300 次左右，如表 1 所示，表 1 是某档案馆自 2021 年 1 月份至 7 月上线运行该系统后的信息共享利用接口每月调用的数量与原有档案信息化系统信息共享利用的对比。

Table 1. Comparison of interface calls for information sharing and utilization in an archive (unit: 10,000 times)
表 1. 某档案馆信息共享利用接口调用对比(单位: 万次)

时间	2022.2	2022.3	2022.4	2022.5
原有档案信息化系统调用次数	0.1	0.1	0	0
该系统使用后调用次数	3.5	4.1	3.8	4.2

通过对比分析表 1 的数据, 自 2022 年 2 月份试运行该系统后, 信息共享利用接口调用次数达到了日均 1300 次, 较为改善各级档案部门之间信息共享利用低的弊端, 提高了用户查档的效率。

5. 系统实现与测试

5.1. 系统实现

档案信息资源共享服务系统的基本架构采用基于 Spring Boot 框架和 Spring Cloud 架构。其中, Spring Boot 用以迅速搭建微服务, Spring Cloud 提供的一系列微服务组件用以统一管理系统中所有微服务注册信息。项目的体系结构为 B/S, 主要开发语言选用 Java8, 数据库选用 MySQL5.6, 前端页面采用 Vue 框架进行开发, 用来部署前端项目的服务器中间件选用 Tomcat8.0 [4]。部分功能界面如图 10~13 所示。



Figure 10. Information release interface
图 10. 资讯发布界面

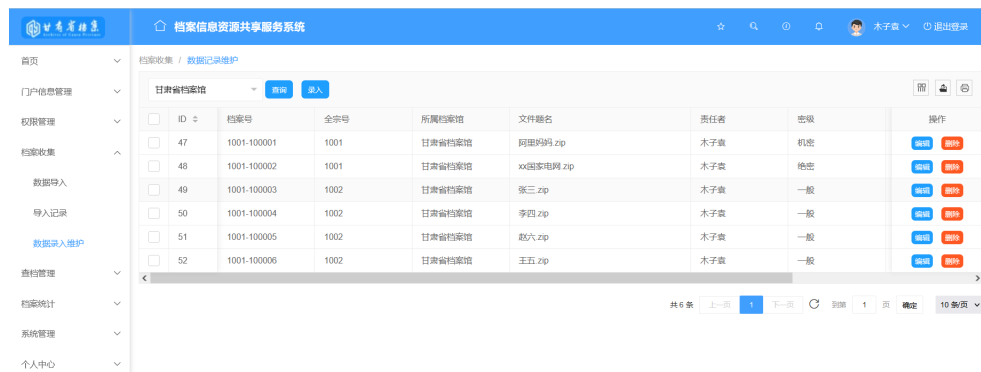


Figure 11. Data import interface
图 11. 数据导入界面



Figure 12. Borrowing utilization statistics interface

图 12. 借阅利用统计界面



Figure 13. File area statistics interface

图 13. 档案区域统计界面

5.2. 系统测试

系统在正式投入使用之前需要进行详细的系统测试，系统测试的主要目的是验证系统是否满足基本的功能需求，设计是否合适以及是否符合真实的业务逻辑。适度的对系统的各个模块进行全面的测试，及时的发现问题，并修复，确保系统的整体的性能。

1) 用户账号注册和登录功能测试

用户功能模块最基本的功能是为档案馆管理员和普通工作用户登录提供入口，需要满足用户账号创建和登录表单校验的需求，通过设计测试用例验证此项功能。如表 2 所示。

2) 资讯发布功能测试

资讯发布功能测试的主要目的是验证资讯的新增、编辑、发布等功能进行验证，通过设计用例验证这些功能是否满足所预期的需求。如表 3 所示。

3) 查档登记功能测试

查档登记功能模块的测试主要包括检索档案基本信息、填写登记表单、受理查询相关功能性的测试。如表 4 所示。

Table 2. User account creation and login function test case
表 2. 用户账号创建和登录功能测试用例

功能描述		用户账号创建以及登录		
用例目的		验证基础模块的功能性		
前提条件		系统中不存在即将用来测试创建和登录的用户名		
用例 ID	输入	期望结果	实际情况	是否一致
YHCJ-001	用户名 zhangsan001	用户名可以使用	该用户名没有被创建过, 可以使用	是
YHCJ-002	用户名 zhangsan001 密码: zs123456 确认密码: zs123456	提示创建成功	用户创建成功	是
YHCJ-003	用户名: z 密码: zs123456 确认密码: zs123456	创建失败并提示原因	用户名长度必须 2 到 10 个字符!	是
YHCJ-004	用户名 zhangsan001 密码: zs123456 确认密码: 123456	创建失败并提示原因	两次输入密码不一致	是
YHCJ-005	用户名 zhangsan001 密码: zs123456 确认密码: zs123456	创建失败并提示原因	该用户名已存在	是
YHDL-001	用户名 zhangsan001 密码: zs123456	登录成功	登录成功	是
YHDL-002	用户名 zhangsan001 密码: 123456	登录失败	用户名或密码错误	是

Table 3. Information release test cases
表 3. 资讯发布测试用例

功能描述		新增资讯、资讯查询以及资讯的发布和下线		
用例目的		对资讯管理模块的功能性进行验证		
前提条件		登录后台管理系统 - 管理员		
用例 ID	动作	期望结果	实际情况	是否一致
GGFB-001	在资讯发布模块, 输入资讯标题、发布单位、发布时间以及资讯内容	添加成功并有相关提示	资讯添加成功提示	是
GGFB-002	公告管理界面或者考试中心网站, 在对应位置输入标题, 点击查询	显示对应输入标题的公告内容	正确显示输入标题的资讯	是
GGFB-003	资讯管理界面, 选择待发布的资讯, 点击发布	显示发布成功	成功修改的资讯状态, 变为已发布	是
GGFB-004	资讯管理界面, 选择已发布的资讯, 点击下线	显示下线成功	成功修改的资讯状态, 变为已下线	是

Table 4. Test case of file checking and registration function**表 4.** 查档登记功能测试用例

功能描述		档案检索、查档登记及受理查询			
用例目的		验证查档登记模块的功能性			
前提条件		登录后台管理系统 - 管理员			
用例 ID	动作	期望结果	实际情况	是否一致	
CDDJ-001	在档案检索页面, 输入档案文件题名, 点击查询	查询出指定文件题名的档案记录	列表展示指定文件提名的档案信息	是	
CDDJ-002	点击指定档案记录后面的查档登记, 填写申请人和档案利用相关信息, 点击确认提交表单	提交成功并提示登记成功	提交成功, 并跳转至受理查询页面	是	

6. 总结

本文以区域性档案信息资源共享服务工作需求为基础, 从解决档案收集、登记、查档、统计等问题出发, 完成了档案信息资源共享利用服务系统的分析、设计和实现, 最后完成了系统功能测试工作。本文采用 Spring Cloud、Vue 等相关技术, 设计并实现了基于 Spring Cloud 框架的档案信息资源共享服务系统。该系统经过测试和试用运行稳定, 主要功能使用良好, 满足档案部门管理需求, 可以实现档案管理便捷化、信息化及可视化的全过程控制, 对于档案领域具有较大的使用价值。

该系统具备较好的经济性、先进性与实用性, 建立区域性的档案共享服务系统可以避免各级档案管理部门建设同类系统, 减少总体政府投资, 防止档案信息资源多次利用, 减轻群众查阅档案负担。先进的信息网络技术, 打破地域限制, 为广大档案利用者提供便捷化、信息化的区域内“异地查档、跨馆服务”, 更好地体现档案馆为社会公众提供的查阅利用服务。该系统投入运行过程中收集的档案资料 and 用户查档数据, 可以运用大数据技术、数据挖掘技术对各种数据进行筛选、分析, 对于智慧档案馆的建设以及档案科研工作提供强有力的数据分析支持。

下一步, 计划从自然语言处理(NLP)、机器学习、知识图谱等人工智能技术出发, 通过构建智能档案知识库、增加意图识别及实体识别模块这两方面来进一步完善系统, 提升用户的使用体验, 使本系统具有更大的社会效益与应用价值。

基金项目

国家档案局科技计划项目“基于大数据智能驱动的档案信息资源挖掘与共享利用服务研究”(2021-X-56)、甘肃省青年自然科学基金计划项目(21JR1RA21)、中央高校基本科研业务费专项(31920210017)。

参考文献

- [1] 杨秦. 基于微服务架构的云平台服务端的设计与实现[D]: [硕士学位论文]. 成都: 电子科技大学, 2020.
- [2] 吴旭君. 基于 Spring Cloud 微服务架构的数据服务系统的实践[J]. 计算机产品与流通, 2020(3): 1.
- [3] 宋永鹏. 基于 MySQL 的数据库查询性能优化[J]. 电子设计工程, 2021, 29(12): 5.
- [4] 徐鹏涛. 基于 Vue 的前端开发框架的设计与实现[D]: [硕士学位论文]. 济南: 山东大学, 2020.
- [5] 柴青山. 基于 MVVM 模式的 Vue.js 框架在物流软件自动化测试系统中的应用研究[D]: [硕士学位论文]. 北京: 北京邮电大学, 2019.

- [6] 杨妍. 基于 Spring Boot 与 Vue 的系统管理模块开发探究[J]. 电声技术, 2019, 43(2): 37-39.
- [7] 熊永平. 基于 SpringBoot 框架应用开发技术的分析与研究[J]. 电脑知识与技术, 2019(36): 2.
- [8] 牛荣, 陈纪龙, 杜义君. 数据库设计中 ER 模型设计的一些基本问题探讨[J]. 信息技术与信息化, 2019(7): 189-192.