

基于SpringBoot的校医院体检预约后台管理系统的的设计

乔 斌, 隋首钢

山东建筑大学, 信息与电气工程学院, 山东 济南

收稿日期: 2021年10月7日; 录用日期: 2021年11月29日; 发布日期: 2021年12月7日

摘 要

随着新冠疫情肆虐全球, 学生对于个人的身体健康越来越重视, 对健康认识不断提高, 健康检查作为防患于未然的主动预防措施在人们思想中已有很深的认识, 校医院体检预约后台管理系统致力于建立一座健康管理机构和学生之间的灵活沟通的桥梁, 该系统采用SpringBoot框架和MyBatis-Plus框架搭建系统业务架构, 前端采用Element UI + Vue技术设计页面, 并利用了JavaScript技术来完善前端页面的布局和人机交互式操作。同时利用了ZooKeeper、Dubbo等技术栈对体检项、体检组和体检套餐等多种信息进行数字化管理, 并使用了Spring Security框架检验院方用户的权限管理的设计, 最终实现院方发布、修改、删除体检相关信息, 为学生提供一个健康的体检方案。

关键词

体检预约后台管理系统, SpringBoot框架, MyBatis-Plus框架

Design of Background Management System of Physical Examination Reservation in School Hospital Based on SpringBoot

Bin Qiao, Shougang Sui

School of Information and Electrical Engineering, Shandong Jianzhu University, Jinan Shandong

Received: Oct. 7th, 2021; accepted: Nov. 29th, 2021; published: Dec. 7th, 2021

Abstract

As the COVID-19 pandemic raging the world, students pay more and more attention to personal

health, and their understanding of health is constantly improving. The Health check is a proactive measure to nip things in the bud, this kind of thought has had a deep understanding in people's minds. The physical examination reservation management system of the school hospital is committed to building a bridge of flexible communication between health management institutions and students. The system uses SpringBoot and MyBatis-Plus frameworks to build the system business architecture, the front-end uses Element UI + Vue technology to design the page, and uses JavaScript technology to improve the layout of the front-end page and man-machine interactive operation. At the same time, ZooKeeper, Dubbo and other technical stacks are used for digital management of physical examination items, physical examination groups, physical examination setmeal and so on, and the Spring Security framework is used to test the design of the hospital user permission management. Finally, the hospital will release, modify and delete information related to physical examination, and provide a healthy physical examination program for students.

Keywords

Physical Examination Appointment Background Management System, SpringBoot Framework, MyBatis-Plus Framework

Copyright © 2021 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

目前医院体检预约管理后台系统是面向广大市民的, 虽然体检项目多, 通用性高, 但是没有任何一款系统是面向在校学生的, 这就导致现有的医院体检预约管理系统面向的用户不专一。本文设计的校医院体检预约后台管理系统是专门为在校生设计的一款体检预约系统, 不仅为学生提供了各种体检组项目, 还为学生量身设计了体检套餐供学生选择, 学生可以自主的根据时间安排进行线上体检预约, 避免了校外医院体检预约的时间限制、人数限制、预约不上的弊端。

2. 系统可行性分析

2.1. SpringBoot 框架

SpringBoot 是由 Pivotal 团队设计开发的高效率框架结构, 通过简化配置来优化 Spring 应用的搭建和开发过程[1] [2]。其中最重要的两个核心就是开箱即用和约定大于配置, 开箱即用是 SpringBoot 之所以被广泛使用的主要因素之一, 只需配置 pom 文件即可直接使用对应功能。这大大简化了编程人员的配置工作, 使他们可以将更多精力投入到项目的功能开发上。后者是约定大于逻辑, 这表明 SpringBoot 本身会提供默认的配置文档, 若其自带的配置符合编程人员的要求, 那么我们就无需再去做相应更改[3] [4]。另外 SpringBoot 中的 Spring Security 框架是一款用于系统用户权限管理的安全框架, 本系统中在对院方不同的用户进行权限管理的时候只需要在当前的工程模块中引入安全框架的相关依赖, 简单配置采用注解式开发就能够实现不同用户的权限管理功能。

2.2. MyBatis-Plus 框架

MyBatis-Plus 是 MyBatis 的增强版, 在 MyBatis 的基础上只对功能做了增强, 无修改, 无侵入, 损耗小, 是一款十分优秀的持久层 ORM (Object Relational Mapping, 对象关系映射)框架[5]。MyBatis-Plus 框

架里面内置了大量的 CRUD (泛指增、删、改、查)操作, 能够满足几乎所有的数据库语言的请求。本系统中只需要在父工程的 pom 文件中引入 MyBatis-Plus 框架依赖并通过内置的 Mapper 接口和实现相应的 Service 就可以实现对数据的增、删、改、查等操作。而且 MyBatis-Plus 内置了分页相关插件, 支持多种数据库语言, 兼容性好, 可靠性高。

2.3. Element UI + Vue 框架

Element UI 是一个为 Vue 而生的 UI 框架, Element UI 是一个 ui 库, 它不依赖于 Vue, 是现阶段和 vue 配合做项目开发的一个较好的 UI 框架[6]。开发人员无需自己定义网页的标签、按钮等, 只需要引入相关的 JavaScript 插件和 Element UI 的组件库即可快速完成网站页面的开发。利用 Element UI 技术对前端界面元素进行设计, 可以有效解决计算机能够显示完整的网页界面而手机端出现网页界面不全的问题[7]。Vue 是一套构建用户界面的渐进式框架[8] [9], 其最大的特点就是数据的双向绑定功能特别强大, 该框架采用 MVVM (Model-View-ViewModel)开发模式, 这种模式将前端从原始的“DOM”操作中解放出来, 我们不再需要花费大量的时间来维护视图和数据的统一, 只需要关注数据“Data”的变化[10]。ViewModel 会随着 View 的变化发生自动更新, 与此同时它的变化也会在 View 中显示出来, 而不需要开发人员为数据绑定投入更多的精力, 可以多注重于业务层的代码实现。Element UI + Vue 框架的组合式开发, 能够让网站页面设计的更加快速、美观、细腻丝滑, 能够给用户很好的用户体验和视觉感受, 大大减少了开发人员在前台所耗费的时间。

3. 系统功能架构

本系统采用微服务框架进行搭建和开发, 选择 Dubbo 服务框架实现服务的治理功能。Dubbo 是阿里巴巴公司开源出来的 SOA 服务化治理方案的核心框架, 它是一个分布式的专注于提供高性能和透明化的 RPC 远程服务调用, 以及 SOA 服务治理的方案[11] [12]。

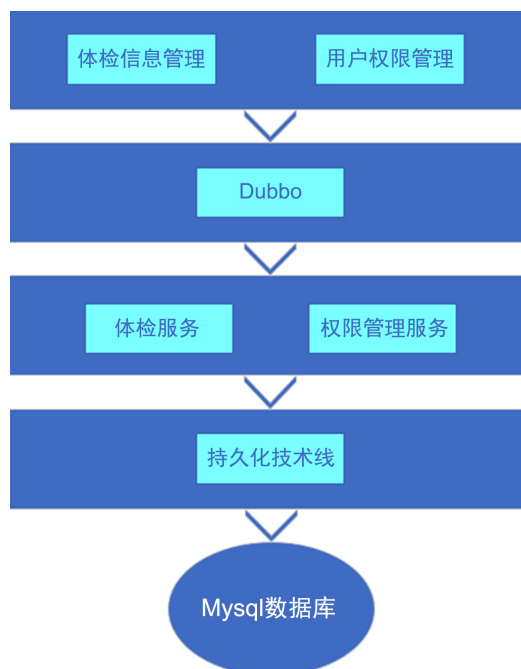


Figure 1. Functional architecture diagram
图 1. 功能架构图

该系统中需要启动服务提供者的服务, 体检和权限管理相关服务全部注册到 Dubbo 的注册中心去, Zookeeper 实现服务的注册和负载均衡功能, 同时监控中心实现对服务的实时监控, 负责统计服务调用次数和服务的运行状态。服务消费者启动时向注册中心订阅自己所需要的服务, 后台根据业务实际需要直接从 Dubbo 中拉去服务使用, 业务完成后通过持久化技术栈将数据保存到数据库中去。该系统的功能架构如图 1 所示。

4. 系统设计

整个系统采用 SpringBoot 框架进行搭建, 系统的接口、实现类、控制层、实体类等代码采用 MyBatis-Plus 代码生成器生成, 然后将生成不同业务类型的代码复制到该系统不同业务的目录下。系统采用 maven 分模块开发方式, 整个系统模块拆分为若干 maven 工程, 父工程提供统一的依赖规范, 而在其余的子工程中存放特定类型的代码。该系统的项目结构简单描述如下:

Parent: 父工程, 打包方式为 pom, 统一锁定依赖的版本, 同时聚合其他子模块便于统一执行 maven 命令。该模块不需要写任何代码, 只需要导入子工程所需要的公共依赖和指定版本供子工程使用;

POJO: 数据库对应的实体类, MyBatis-Plus 代码生成器根据数据库中的表生成相对应的实体类, 然后将实体类代码全部复制到该目录下;

Utils: 工具类模块, 该目录存放各种封装好的工具类供开发人员使用, 存放项目中使用到的一些工具类、返回结果和常量类;

Interface: 该目录主要存放服务的接口, 该接口只起到声明作用, 具体的业务需要在该接口的实现类中编写;

Provider: Dubbo 服务方模块, 存放服务接口的实现类、Dao 接口、Mapper 映射文件等, 作为服务提供方, 注册到 Dubbo 的注册中心去, 供其他服务调用。所有体检相关信息的业务全部在本模块的实现类中编写;

Controller: Dubbo 消费方模块, 也称为控制层, 存放 Controller、HTML 页面、JavaScript、CSS、Spring 配置文件等, 作为服务消费者同样需要注册到 Dubbo 的注册中心去, 该模块处理前台的 Http 请求, 根据具体的请求来调用 Provider 中的不同的业务, 然后将业务处理结果返回到前台;

Security: 权限管理模块, 该模块主要负责系统用户的权限管理的设计。

4.1. 数据库设计

数据库位于整个系统的底层, 是整个系统设计中最为重要的一环。该系统的数据库总共包括 10 张表, 包括体检相关信息的 5 张表和用户权限管理的 5 张表。主要数据表的简单描述如下:

t_checkitem, 体检项信息表, 用于保存院方发布的各种体检项信息, 其表中字段描述如表 1 所示:

Table 1. Physical examination items information sheet
表 1. 体检项信息表

列名	数据类型	含义	备注
id	int	体检项 id	主键
code	varchar (16)	体检项编码	
name	varchar (32)	体检项名字	
sex	char (1)	体检项适合的性别	
age	varchar (32)	体检项适合的年龄	

Continued

price	float	体检项价钱
type	char (1)	体检项类型, 包括检验和检查
remark	varchar (128)	体检项描述
attention	varchar (128)	体检项注意事项

t_checkgroup: 体检组信息表, 用于保存体检组相关的信息, 该表和体检项信息表是多对多关系, 一个体检组可以包括多个体检项, 同时一个体检项也可以隶属于多个体检组, 二者通过关联表产生关联关系, 其表中的字段描述如表 2:

Table 2. Physical examination group information sheet

表 2. 体检组信息表

列名	数据类型	含义	备注
id	int	体检组 id	主键
code	varchar (32)	体检组编码	
helpCode	varchar (32)	体检组助记码	
name	varchar (32)	体检组名字	
sex	varchar (32)	体检组适合的性别	
remark	varchar (128)	体检组描述	
attention	varchar (128)	体检组注意事项	

t_setmeal: 套餐信息表, 用于保存院方发布的各种套餐信息, 其表中字段描述如表 3 所示:

Table 3. Setmeal information sheet

表 3. 套餐信息表

列名	数据类型	含义	备注
id	int	套餐 id	主键
name	varchar (128)	套餐名字	
code	varchar (8)	套餐编码	
helpCode	varchar (32)	套餐助记码	
sex	char (1)	套餐适合的性别	
age	varchar (32)	套餐适合的年龄	
price	float	套餐价格	
remark	varchar (128)	套餐描述	
attention	varchar (128)	套餐注意事项	
img	int	体检项 id 对应的外键	外键

t_user: 用户信息表, 用于存储登录用户的相关信息, 其表中的字段描述如表 4 所示:

Table 4. User information sheet

表 4. 用户信息表

列名	数据类型	含义	备注
id	int	用户 id	主键
user name	varchar (32)	用户姓名	
password	varchar (256)	用户密码	
gender	varchar (1)	用户性别	
birthday	data	用户生日	
remark	varchar (32)	用户描述	
station	varchar (1)	用户状态	
telephone	varchar (11)	用户手机号码	

t_role: 角色信息表, 用户存储用户所拥有的角色的相关信息, 其表中的字段描述如表 5 所示:

Table 5. Role information sheet

表 5. 角色信息表

列名	数据类型	含义	备注
id	int	角色 id	主键
name	varchar (32)	角色名称	
keyword	varchar (64)	角色关键字	
description	varchar (128)	角色描述	

t_menu: 权限信息表, 用户存储角色所拥有的权限的相关信息, 其表中的字段描述如表 6 所示:

Table 6. Permission information sheet

表 6. 权限信息表

列名	数据类型	含义	备注
id	int	权限 id	主键
name	varchar (128)	权限名称	
linkUrl	varchar (128)	权限相关的页面	
path	varchar (128)	权限所在的路径	
priority	int (11)	权限优先级	
icon	varchar (64)	权限关键字	
description	varchar (128)	权限描述	
parentMenuId	int (11)	父权限 id	
level	int (11)	权限层次	

而最终设计好的数据库架构如图 2 所示:

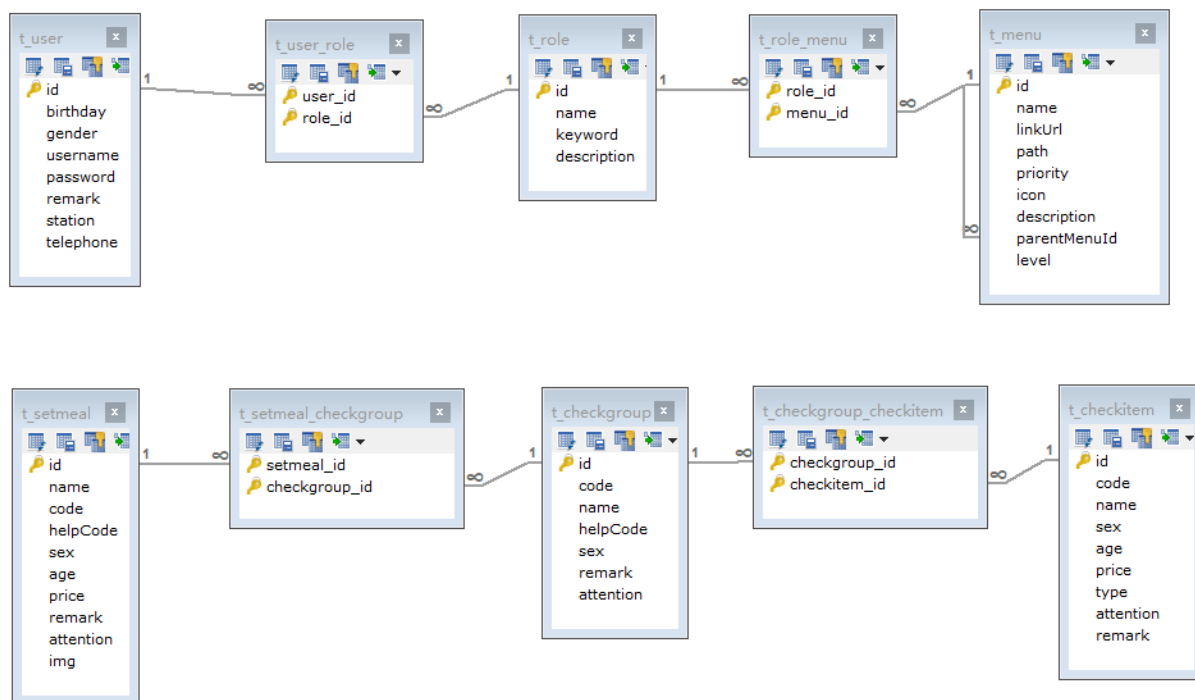


Figure 2. Database architecture diagram

图 2. 数据库架构图

4.2. 体检项模块设计

系统默认显示的页面是体检项页面, 该页面显示各种体检项信息。院方用户登录后可以查看当前所有的已发布体检项相关信息, 可以输入搜索条件查询指定的体检项。院方能够点击修改按钮对已经发布的体检项信息进行修改, 能够点击删除按钮删除不需要的体检项, 也能够根据实际需要新建一个体检项。系统首页体检项信息的展示如图 3 所示:

序号	项目编码	项目名称	适用性别	适用年龄	项目说明	操作
1	0001	身高	不限	0-100	身高	编辑 删除
2	0002	体重	不限	0-100	体重	编辑 删除
3	0003	体重指数	不限	0-100	体重指数	编辑 删除
4	0004	收缩压	不限	0-100	收缩压	编辑 删除
5	0005	舒张压	不限	0-100	舒张压	编辑 删除
6	0006	裸视力(右)	不限	0-100	裸视力(右)	编辑 删除
7	0007	裸视力(左)	不限	0-100	裸视力(左)	编辑 删除
8	0008	矫正视力(右)	不限	0-100	矫正视力(右)	编辑 删除

Figure 3. Display of physical examination items

图 3. 体检项展示

4.3. 体检组模块设计

用户可以点击前台页面导航栏的体检组信息管理按钮实现对体检组信息的查看，体检组信息展示页面如图 4 所示：



Figure 4. Display of physical examination group
图 4. 体检组展示

用户可以查看所有体检组信息，也可以输入搜索条件查询指定的体检组信息。一个体检组包括了多个体检项，因此对于体检组的增、删、改操作不仅要操作体检组本身的数据，还需要对当前体检组下属的体检项信息进行操作。

体检组新增：用户点击新建按钮，弹出对话框输入相关的体检组信息，同时也要回显当前院方发布的所有的体检项相关信息供用户选择，选择当前新建体检组所需要的体检项选项，完成体检组信息的创建，体检组新建展示和体检项信息回显展示如图 5 和图 6 所示：

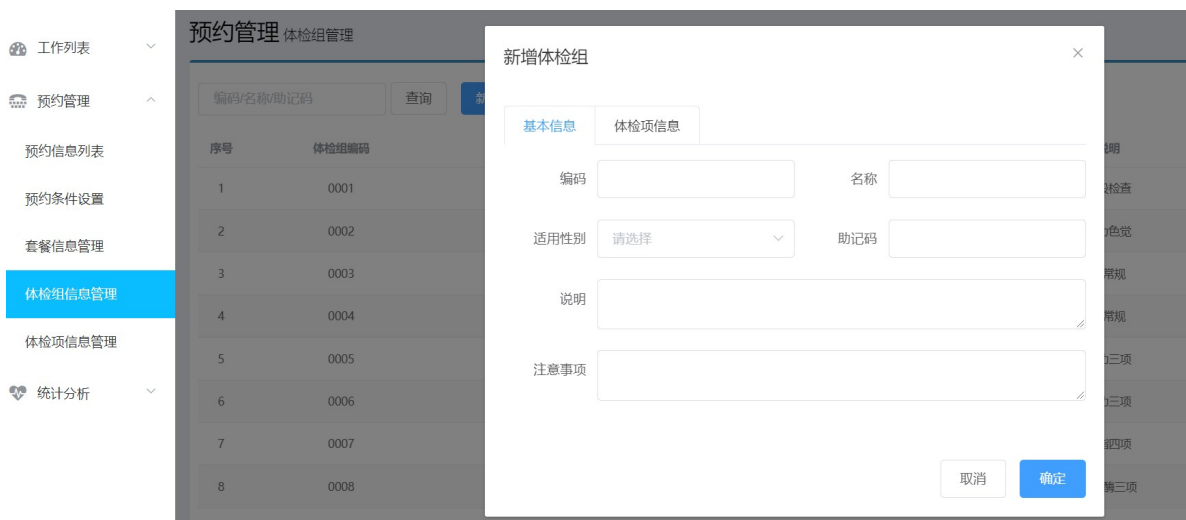


Figure 5. Display of the echo physical examination items
图 5. 体检项回显展示



Figure 6. Display of physical examination items information
图 6. 体检项信息回显

体检组编辑：用户点击编辑按钮，实现对体检组信息的编辑。点击按钮的同时要实现当前正在编辑的体检组信息的回显和体检项信息的回显，然后在对体检组的进行修改，修改完成点击提交按钮即可完成数据库中体检组信息的更新。体检组编辑展示如图 7：



Figure 7. Display of editing physical examination group
图 7. 体检组编辑展示

体检组删除：用户点击删除按钮，可以实现体检组的删除。体检组和体检项是通过关联表产生关联关系的，数据库中删除体检组的同时也要把关联表的数据删除。

4.4. 套餐模块设计

该系统的套餐模块是专门为学生设计的，是本系统中的特色模块，该模块下包含多种套餐，可供不同群体的学生选择和体检，其中包括为毕业生设计的男女入职体检套餐和针对新生设计的全身体检套餐

等。套餐与体检组是多对多关系，套餐下面包括多个体检组，学生预约套餐成功后即可根据该套餐的信息去相应的门诊去体检。用户可以点击前台页面导航栏的体检组信息管理按钮实现对体检套餐的查看，体检套餐信息展示页面如图 8 所示：



Figure 8. Display of setmeal
图 8. 套餐展示

套餐查询：院方可以根据实时情况查询所需要的套餐信息，直接在输入框输入套餐编码、套餐名称或者套餐助记码即可查询其对应的套餐。查询编码为 001 的套餐展示如图 9 所示：



Figure 9. Display of querying setmeal
图 9. 套餐查询展示

套餐新增：院方用户点击新建按钮，弹出对话框，输入套餐相关信息，包括套餐编码、名称、适用性别和助记码等，新增套餐的同时也要回显当前院方发布的所有的体检组相关信息供用户选择，输入完套餐信息和选择完该套餐下的体检组后点击确定按钮即可实现套餐的新增。新增套餐和体检组信息回显如图 10 和图 11 所示：

套餐编辑：用户点击编辑按钮，实现对套餐信息的编辑。点击按钮的同时要实现当前正在编辑的套餐信息的回显和体检组信息的回显，然后再对当前套餐进行修改，修改完成点击确定按钮即可完成数据库中套餐数据的更新。编辑编码为 0001 的套餐如图 12 所示：



Figure 10. Display of creating setmeal
图 10. 套餐新增展示



Figure 11. Display of the echo setmeal
图 11. 套餐回显展示

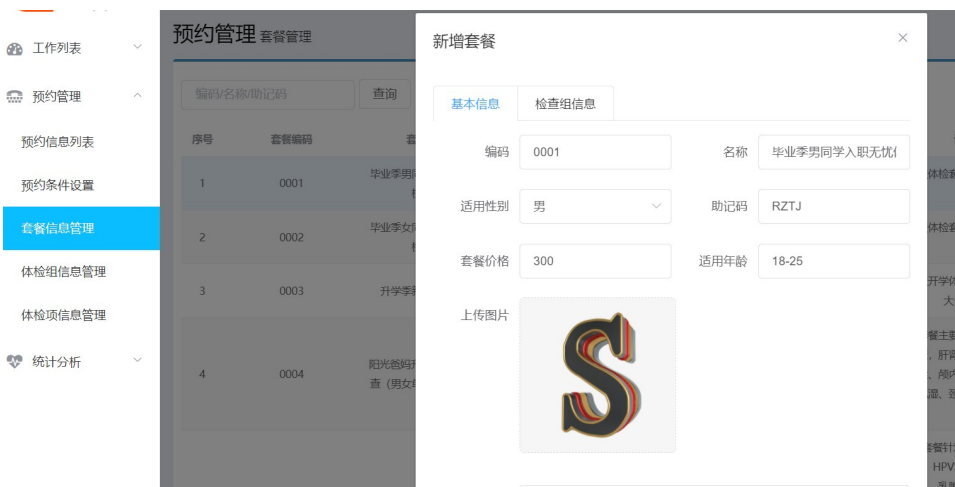


Figure 12. Display of editing setmeal
图 12. 套餐编辑展示

套餐删除: 用户点击删除, 弹出对话框二次确认是否删除, 点击确认完成数据库中套餐数据的删除。套餐和体检组是通过关联表产生关联关系的, 删除数据库中的套餐数据同时也要把关联表的数据删除。

4.5. 用户权限管理设计

该系统中根据不同的用户设置了不同的权限, 院方不同级别的用户拥有不同的操作权限, 主任、副主任医师等具有对体检组、体检项和套餐的增、删、改、查权限, 普通医生只有查看体检相关信息权限。权限管理设计需要五张表, 即 `t_user` (用户信息表)、`t_user_role` (用户和角色关联表)、`t_role` (角色表)、`t_role_menu` (角色和权限关联表)和 `t_menu` (权限表)。权限管理模块不同于其它工程模块, 关于权限管理的业务需要写在 `SpringSecurity` 安全框架中的 `UserDetailsService` 接口的实现类中, 该接口是 `Spring Security` 专门用于权限认证的接口, 其实现类的返回值类型是 `UserDetails`。权限管理的具体设计如下阐述:

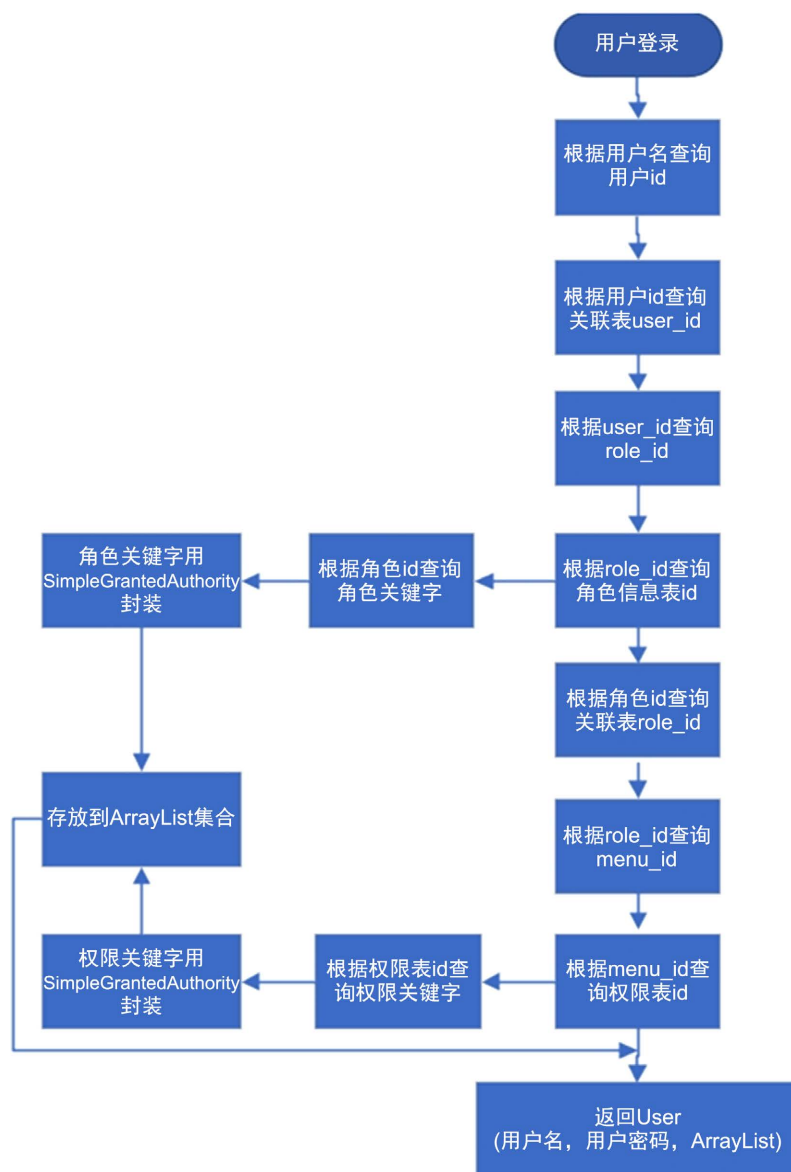


Figure 13. Flow chart of permission management
图 13. 权限管理流程图

首先用户通过用户名和密码登录系统, 根据用户名去数据库中的 `t_user` 表查询该用户的信息, 根据查询的用户 `id` 去 `t_user_role` 关联表中查询该用户 `id` 对应的外键所对应的角色 `id` 对应的外键, 然后再根据 `t_user_role` 关联表中查询出来的角色 `id` 对应的外键去 `t_role` 表中查询该用户所拥有的角色 `id`, 进而查询到该用户所拥有的角色信息, 将角色信息中的角色关键字用 Spring Security 中专门的权限认证构造函数 `SimpleGrantedAuthority` 封装, 封装完后在将结果存放到 `ArrayList` 集合中。然后采用相同的方式将查询到的权限关键字用 `SimpleGrantedAuthority` 封装, 封装完后同样存放到 `ArrayList` 集合中去。权限管理业务最终的返回值类型为 `User`, 其中有三个参数, 第一个为系统登录用户的用户名, 第二个为用户密码, 第三个为存放着用户角色和权限的 `ArrayList` 集合。具体的业务实现流程如图 13 所示。

具体的权限认证方式在 Controller 层上采用注解式开发进行权限认证, 这里使用 `@PreAuthorize` (“`hasAnyAuthority` (‘权限关键字’)”)和 `@Secured` (“角色关键字”)这两个注解实现。`@PreAuthorize` (“`hasAnyAuthority` (‘权限关键字’)”)表示当前登录用户拥有指定权限关键字才能够调用业务层方法, `@Secured` (“角色关键字”)表示当前用户是指定角色才能调用业务层方法。

5. 结语

随着计算机科学技术在校园生活中的应用, 以信息化的方式来代替学生们线下体检预约必将成为未来校园的发展趋势。本文根据当前疫情的扩散和现有的医院体检预约管理后台系统不够专一的情况, 提出了一种基于 SpringBoot 的校医院体检预约后台管理系统的设计。该系统为不同群体的学生提供了相对应的体检套餐, 学生们能够根据自己的性别、年龄和实际情况选择适合自己的体检套餐进行预约体检。由于本系统是面向在校生的, 不为社会人员开放, 因此能够大大提高学生体检流程的效率, 减少学生用户体检排队时长, 既加强了学生和校医院之间的联系, 也进一步提升了学生对校医院的满意度, 为其它校医院的体检预约工作提供了参考。

基金项目

2020 年国家自然科学基金, 62003191, 基于稀疏慢特征分析的间歇过程故障诊断方法研究。

2020 年山东省自然科学基金, ZR2020QF072, 基于融合特征稀疏表征 GSFA 的间歇过程故障检测和识别方法研究。

参考文献

- [1] 刘子凡, 郭昱君. 基于 SpringBoot+MyBatis 的个人博客系统设计与实现[J]. 现代信息科技, 2021, 5(8): 104-107+111.
- [2] 刘超慧, 杨雨涵, 等. 基于 SpringBoot 的教学资源平台设计与实现[J]. 科技风, 2021(11): 92-93.
- [3] 钟怡昀, 郭昱君. 基于 SpringBoot 的博客管理系统设计与实现[J]. 现代信息科技, 2021, 5(7): 18-20+24.
- [4] 李娟, 彭剑. 基于 SpringBoot 的高校课程知识库的设计与实现[J]. 电脑知识与技术, 2021, 17(6): 70-71.
- [5] 周旺, 吴昌平, 李贤和. 师生双选系统后端设计[J]. 无线互联科技, 2021, 18(12): 45-46.
- [6] 王志文. Vue+Elementui+Echarts 在项目管理平台中的应用[J]. 山西科技, 2020, 35(6): 45-47.
- [7] 饶平平, 沈益, 朱清鹤, 等. 工程信息精细化管理平台架构设计研究[J]. 土木工程信息技术, 2021, 13(2): 27-35.
- [8] 罗宇瑞, 张秋霞. “互联网+”背景下校园闲置物品交易平台设计研究[J]. 河南科技, 2021, 40(5): 30-32.
- [9] 黄晟祺, 汪玉筋, 等. 基于前后端交互的就医无忧系统的设计与实现[J]. 信息系统与数据处理, 2021, 24(6): 53-56+49.
- [10] 方生. 基于“Vue.js”前端框架技术的研究[J]. 电脑知识与技术, 2021, 17(19): 59-60+64.
- [11] 赵子晨, 朱志祥, 蒋来好. 构建基于 Dubbo 框架的 SpringBoot 微服务[J]. 计算机与数字工程, 2018, 46(12):

2539-2543+2551.

- [12] 尹伟. 基于 Dubbo 和 Spring MVC 的红十字会业务平台的设计及优化[D]: [硕士学位论文]. 南昌: 东华理工大学计算机技术学院, 2018.