|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | | | |
| 50da81cb39dbb6fdf8b8fff80b24ab18962b3708 | | |  | | | | ○ **A** **基础理论**  ● **B** **应用研究**  ○ **C** **调查报告**  ○ **D** **其他** | | | | | |
|  | | **岭南师范学院** | | | | | |  | | | | |
|  | **本科生毕业论文（设计）** | | | | | | | | |  | | |
|  | | | | | | | | | | | | |
|  | **基于vue+springboot的疫情防控管理系统** | | | | | | | | | | |  |
|  | | | | | | | | | | | | |
|  | | | | **二级学院** | **：** | **文学与传媒学院** | | | | |  | |
| **专 业** | **：** | **计算机科学与技术** | | | | |
| **年 级** | **：** | **2019级** | | | | |
| **学 号** | **：** | **201908344119** | | | | |
| **作者姓名** | **：** | **刘品昭** | | | | |
| **指导教师** | **：** | **熊建芳 讲师** | | | | |
|  | | | |  |  | **陈禹璐 工程师** | | | | |  | |
|  | | | | | | | | | | | | |
|  | | | | **完成日期** | **：** | **2023年 4 月 25日** | | |  | | | |

**基于vue+springboot的疫情防控管理系统**

专业名称：计算机科学与技术

作者姓名：刘品昭

指导教师：熊建芳

**论文答辩小组**

**组长**： 卢利琼

**成员**： 张志洁

卢禹琨

**论文成绩：**

**目 录**

[1 前言 1](#_Toc133398546)

[1.1 研究背景及意义 1](#_Toc133398547)

[1.2 发展现状 2](#_Toc133398548)

[2 相关技术概述 3](#_Toc133398549)

[2.1 前后端分离架构 3](#_Toc133398550)

[2.2 前端开发 3](#_Toc133398551)

[2.3 后端开发 3](#_Toc133398552)

[2.4 数据库开发 4](#_Toc133398553)

[3 系统总体分析 5](#_Toc133398554)

[3.1 可行性分析 5](#_Toc133398555)

[3.2 功能需求分析 5](#_Toc133398556)

[3.3 系统流程分析 6](#_Toc133398557)

[4 系统概要设计 8](#_Toc133398558)

[4.1 系统功能设计 8](#_Toc133398559)

[4.1.1 前台功能模块 8](#_Toc133398560)

[4.1.2 后台功能模块 9](#_Toc133398561)

[4.2 数据库设计 10](#_Toc133398562)

[4.2.1 数据库实体模型设计 10](#_Toc133398563)

[4.2.2 数据库结构设计 14](#_Toc133398564)

[5 系统详细设计与实现 17](#_Toc133398565)

[5.1 系统界面设计 17](#_Toc133398566)

[5.1.1 前台界面 17](#_Toc133398567)

[5.1.2 后台界面 21](#_Toc133398568)

[5.2 后台服务端 27](#_Toc133398569)

[6 系统测试 29](#_Toc133398570)

[7 结论 31](#_Toc133398571)

[7.1 总结 31](#_Toc133398572)

[7.2 展望 31](#_Toc133398573)

**基于vue+springboot的疫情防控管理系统**

作者　刘品昭　指导教师　熊建芳 讲师

(岭南师范学院算机与智能教育学院，湛江 524048)

**摘　要：**每到季节交替之时，是流行传染病的高发季节，如近段时间甲流的阳性率超过新冠，传染性强，病症也较为严重。为了广大师生的身体健康，让校园管理更加高效，故开发疫情防控管理系统。该系统采用前后端分离的架构，前端使用Vue框架进行开发，后端使用SpringBoot框架进行开发。系统的部署和运行需要使用MySQL数据库和Tomcat服务器。本系统主要有师生打卡、物资管理、流感信息图表展示、需关怀人员管理、人员权限管理等功能，目的是帮助学校更好地管理和防控甲型流感等流行疫情，提高应对突发事件的能力和水平。

**关键词：**流行传染病；管理系统；SpringBoot；MySQL

**Epidemic Prevention and Control Management System Based on Springboot+Vue**

Pinzhao Liu Instructor: Jianfang Xiong

School of Computer Science and Intelligence Education, Lingnan Normal University, Zhanjiang, 524048 China

**Abstract:** At the turn of each season, there is a high incidence of infectious diseases. For example, recently, the positive rate of swine flu exceeds that of the new coronavirus, so the infectivity is strong and the disease is relatively serious. For the health of teachers and students and to make campus management more efficient, the epidemic prevention and control management system was developed. The system adopts the architecture of front end separation, the front end uses the Vue framework for development, the back end uses the SpringBoot framework for development. The MySQL database and Tomcat server are required for system deployment and operation. The system mainly has the functions of teachers and students punching in, material management, influenza infographic display, caring personnel management, personnel authority management, etc. The purpose is to help schools better manage and prevent influenza A and other epidemics, improve the ability and level of response to emergencies.

**Key words:** Epidemic diseases; Management System; SpringBoot; MySQL

# 1 前言

## 研究背景及意义

每到季节更替，都会伴随着如甲流等流行传染病，其症状有轻有重，从感冒发烧再到中耳炎、肺炎、心肌炎、脑膜炎等，一定程度上会影响我们的工作生活，甚至是生命健康[1]。而且虽然最近的新冠病毒威力已经逐渐减小，防控疫情的保卫战取得了胜利，但我们高校、企业、政府也存在对突发卫生安全事件的应对经验不足，管理方法落后、信息协作与共享不透明等问题，我们需要认真汲取教训，提高我们应对突发公共卫生事件的应对能力。

同时随着互联网的普及，人们对于互联网设备及其产品的依赖越来越深。其中互联网产品中以Java开发的应用比较常见，市场上对管理系统的开发积累了不少经验，解决方案比较多，其中以 Springboot框架开发的疫情防控管理系统具有开发人员多、开发成本低、开发周期短、市场前景广阔等优点[2]。它解决了现有信息采集的问题如人工采集费用高、大量纸张浪费、数据安全性难以保证等。开发防控系统不仅能符合互联网浪潮趋势，提升疫情防控的管理水平，也能保障信息安全和隐私还有让高校的支出、企业的研发、后期的维护等费用都在一个合理的范围。此外，基于云端技术的疫情防控管理系统还具有实时数据共享、数据分析和可视化等功能，能够为疫情防控决策提供更加准确、科学的依据[3]。

近年来，我国经济发展迅速，电脑逐渐普及，普通人也能接触到高端的电脑设备，此外受教育水平较高的师生群体，日常学习办公、科研学习也离不开电脑设备，对电脑简单操作比较熟悉。因此，本人想开发一个基于SpringBoot+Vue的疫情防控管理系统，师生可以在电脑网页完成相关的操作，简化繁琐的人工采集信息步骤，易于被广大师生所理解、接收和认可。

值得注意的是，该管理系统的开发不仅是一项技术创新，更是一项社会责任和义务。在生命健康面前，各方应该携手合作，共同应对流行传染病挑战。政府、高校、企事业单位等应该加强协作和信息共享，共同推进疫情防控管理系统的开发和应用，为保障人民健康和社会安定做出积极贡献[4]。

## 发展现状

根据之前COVID-19 疫情各个高校防疫方法、管理手段等来看，对于学生健康信息的数据管理与收集主要有以下模式:

(1)纸质资料采集，线下集中汇报。遵循上级管理部门下达的报告指标，使用信息采集自上而下，汇总报告自下而上的方式。因受限于时间、成本、人员数量，主要对重点风险人群采集，采集完成后，对相关指标进行分层计算和上报决策层。

(2)第三方工具采集，线下集中汇报。该模式主要号召学生通过第三方工具，如问卷星，微信群接龙，企业微信等工具收集数据，相关工作人员再自己进行统计、汇总，最后集中上报给决策层。

(3)开发出了校园特色的信息收集小程序。师生可以通过微信小程序打卡来进行个人健康信息的提交与更新，但是相对来说功能较少，只有基本的信息采集，主要由社会上的小企业开发维护，个人信息的保密性与安全性难以保障。

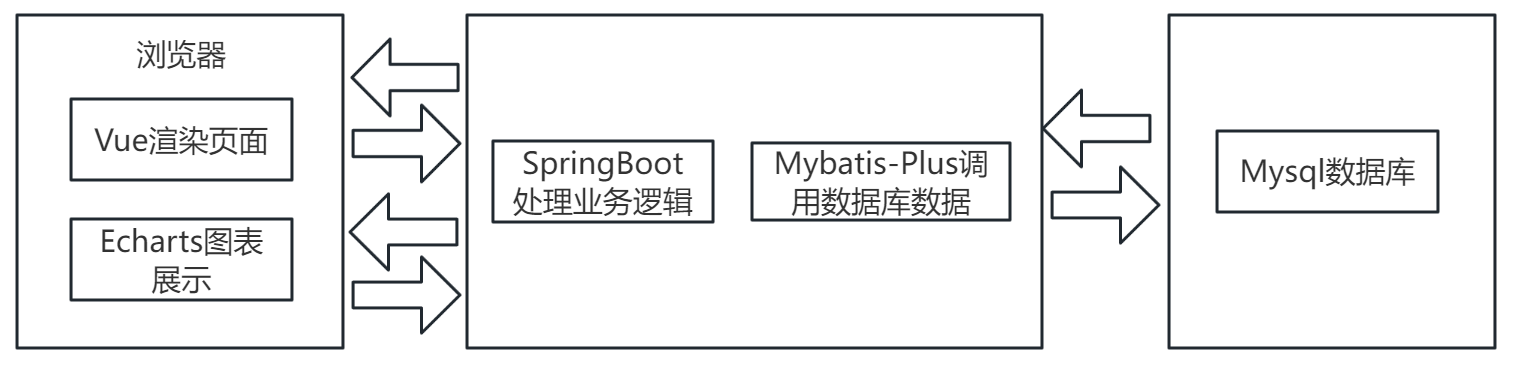
以上三种情况都各具特色，但多多少少都存在一定问题。第一点情况采集依赖手工表格等纸质材料，造成大量的纸张浪费，不利于节约资源，保护环境，不响应国家十四五出台的节能减排相关政策，同时线下手工采集学生基本信息，会造成大量人员聚集，容易交叉感染；第二点情况相对于第一种模式而言，填报效率提高了，统计效率依然低，需要手动统计和汇总数据，最后上报结果。第三种就是第三方平台不稳定，数据不安全，隐患大，容易发生数据泄露，被不法分子利用危害社会,以上都是传统疫情防控存在的突出问题[5]。

鉴于以上情况，本人打算开发设计一种基于Vue+SpringBoot的数据分析的校园疫情防控系统。本文系统的使用不仅对学生自身、老师们来说可以减少很多繁琐的操作，方便快捷地采集后期条件允许，用互联网大公司腾讯或者阿里的服务器，数据库部署系统，可以有效降低社会的小公司窃取个人信息，低价卖给不法分子，损害师生的财产安全的风险，此外还可以平台自动智能计算，减少冗余繁琐的统计步骤，降低学校各个部门之间防控信息获取错误的概率，可以更好更直观地给管理者提供参考。因此，该系统相比上述传统的管理模式，可以给学生、宣传志愿者以及学校提供了更为便捷的管理途径，更有力高效地保护广大学生的师生健康、数据安全。

# 相关技术概述

## 前后端分离架构

前后端分离架构是一种软件架构模式，它将一个应用程序的前端和后端分离开来，使得它们可以独立开发、部署和扩展[6]。在这种架构中，前端开发者负责向后端请求数据库的数据和展示数据，后端开发者则专注于对业务解析从而对数据进行处理，前后端分离架构有让前端和后端可以独立开发，使得开发进度更快、前端和后端可以独立部署和扩展，使得应用程序更加灵活和可维护，可以更好地实现前端性能优化，如减少数据传输量和减少页面加载时间等优点。如下图图1所示。



**图1 前后端分离架构**

## 前端开发

前端部分包括用户界面（UI）和与用户交互的逻辑，如用户输入验证、表单处理、页面导航等。它通常由HTML、CSS和JavaScript等技术构建，也可以用最流行的Vue框架简化开发，Vue是目前时兴和日渐完善的前端开发框架，轻量级、渐进式、组件化是它的特点，它们可以让程序运行在浏览器、移动设备或桌面应用程序上。还将采用ECHARTS图表技术，将师生的健康数据直观地通过柱状图、饼状图、折线图等展现，提高用户体验。

开发工具：Visual Studio Code、Chrome浏览器

## 后端开发

主要技术：SpringBoot、Mybatis-Plus、CentOS。Spring Boot是一个开源的Java应用程序开发框架，旨在简化Spring应用程序的开发、配置和部署。它提供了一组默认的配置，使得开发人员可以快速地构建基于Spring的应用程序，并且可以轻松地集成其他开源库和框架[7]。可以使开发人员在不需要额外配置的情况下快速搭建Web应用程序。Mybatis-Plus是一个用于操作Mysql数据库的持久层，可通过API调用来实现对数据库的增删改查。CentOS是社区企业级的Linux的发行版本之一，稳定性强且应用范围广。

开发工具：Idea、阿里云服务器、Xshell。

## 数据库开发

主要技术：Mysql。Mysql是以文档与集合方式存储数据的非关系型数据库。

开发工具：Navicat 2019。

MySQL是一种免费且开源的关系型数据库管理系统，它具有易于使用和部署、跨平台性、高性能和可伸缩性、可靠性和安全性、开源自由以及大量的支持和文档等优点。MySQL被广泛应用于各种规模的应用程序和网站的开发中，成为一种非常流行的数据库管理系统。开发者可以使用MySQL来存储和管理各种类型的数据，同时使用MySQL提供的丰富的查询和数据处理功能，使得应用程序和网站的开发变得更加高效和简单。

# 系统总体分析

## 可行性分析

本次研究，根据实际的开发需求，主要采用的C/S模式开发，使用Springboot作为后端开发框架、Vue作为前端开发框架，并且使用MySQL数据库，用Apifox来进行接口测试，最后独立部署到云服务上线，让大家可以在网页上访问。

从技术可行性方面来看，Vue是一套用于构建用户界面的渐进式的Js框架，只关注视图层，采用自底向上增量开发的设计，Vue框架采用MVVM模式，实现数据的双向绑定，能将后端传递的数据转化成所看到的页面，也能将所看到的页面转化成后端的数据[8]。它与Anoular Js、React框架相比，Vue更加管单，运行速度更快，易于与第三方库整合[9]。Spring Boot是由Pivotal团队提供的框架，其设计目的是简化Spring应用的初始搭建以及开发过程。将开发人员从繁琐的配置中解放，专心于业务逻辑和系统架构。因此，在技术可行性方面是没问题的。

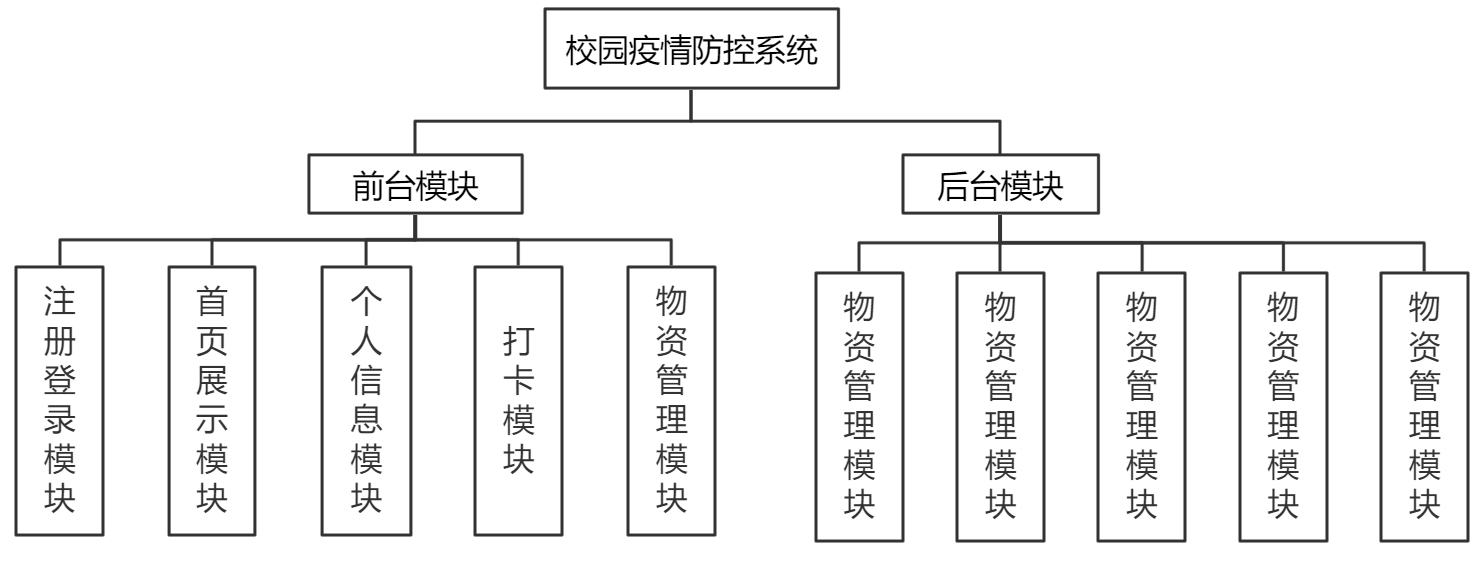
就操作可行性而言，本系统按照之前的分析，师生只需简单操作电脑，完成简单的打卡定位、记录健康信息等操作，再经过点击提交即可被管理员看到，进行高效的管理。对于有一定文化水平的高校师生使用应该不成问题。因此，在可操作性方面是可行的。

就经济可行性而言，本人做为开发者，不用支付给软件公司高额的成本，只需一台有相关开发工具的笔记本电脑，一个阿里云服务器即可，成本是极低的。

经过上述分析可以看出，本系统在技术可行性、操作可行性、经济可行性等方面均表现不错，是可行的。

## 功能需求分析

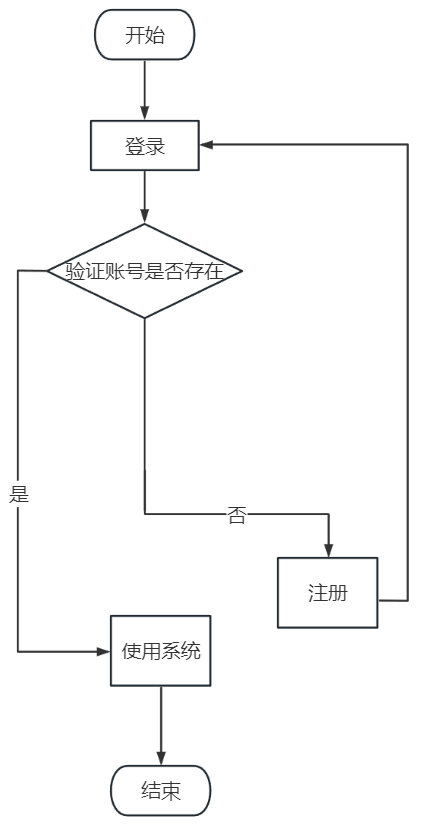
本系统包括了两大模块，前台模块和后台两大模块。其中这两个模块还包含了一些功能模块，如下图2所示。



**图2 系统功能模块图**

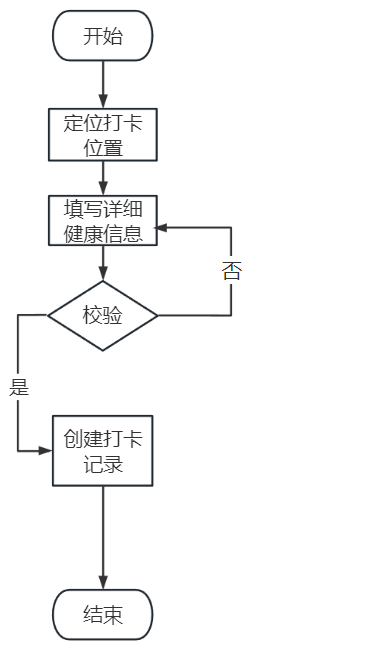
## 系统流程分析

用户登录流程，主要用于验证用户是否能够登录，在表中查不到或者为空就提示一下，然后请用户到注册页面注册，如图3所示。



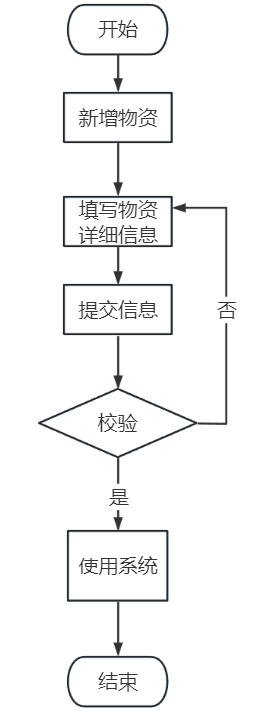
**图3 系统登录流程图**

打卡流程，主要就是用户在填写打卡信息并进行提交时候校验，然后创建打卡记录，如图4所示。



**图4 打卡流程图**

物资记录创建流程，主要就是用户在填写物资信息并进行提交时候校验，然后创建物资记录，如图5所示。



**图5 物资记录创建流程图**

# 系统概要设计

## 系统功能设计

### 前台功能模块

前台模块主要有以下四个模块，如图6所示。

（1）登录注册模块

该模块使用市场流行的session技术来实现前后端分离的登录验证，简单高效的同时，安全性也有所保障，同时也可以实现注册和修改密码功能。

（2）健康数据统计

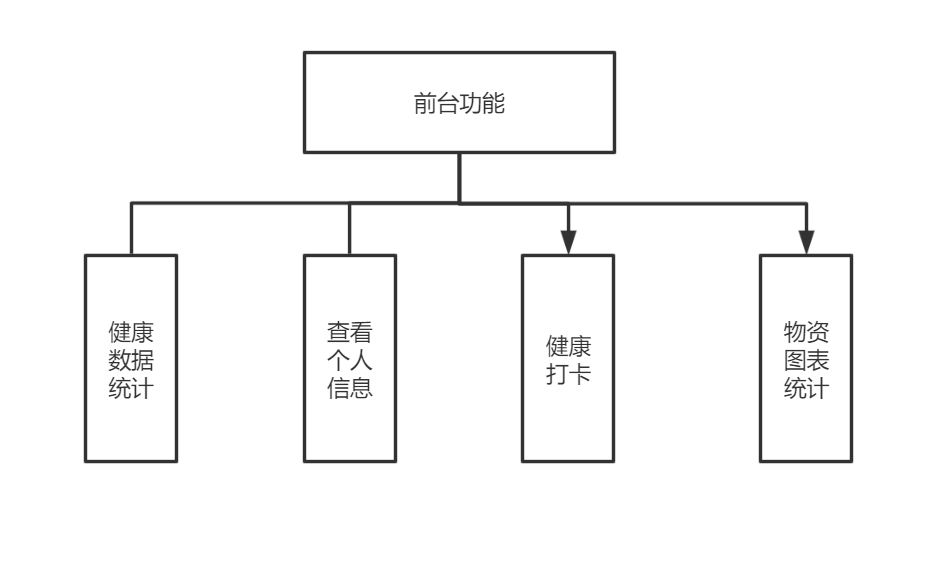
该模块用多图表来展示校园中的健康情况，如发烧人数、感冒人数、健康人数，各级学院的具体健康情况等。

（3）师生打卡模块

学生和教职工每天可以登录网页，进行定位打卡，并填写自己的健康信息，配合预防部门的工作，也可以打完卡后查看自己的个人信息。

1. 物资图表统计

使用丰富的Echarts插件来实现饼状图、柱状图等，直观展示物资情况。



**图6 前后功能模块图**

### 后台功能模块

后台管理功能涉及4个模块，有8个相关功能，如图7和图8所示。

后台管理PC端主要功能模块:

（1）登录模块

     该模块用于提供给管理人员登录，来对疫情防控平台进行后台管理。

（2）学生管理模块

     该模块用于管理学生的健康信息，进行基本的增删改查操作，简化管理流程。

（3）教师管理模块

该模块用于管理教师的健康信息，进行基本的增删改查操作，简化管理流程[10]。

（4）需关怀人员管理模块

该模块用于管理有发病症状的师生，对他们给予及时的关怀和帮助。

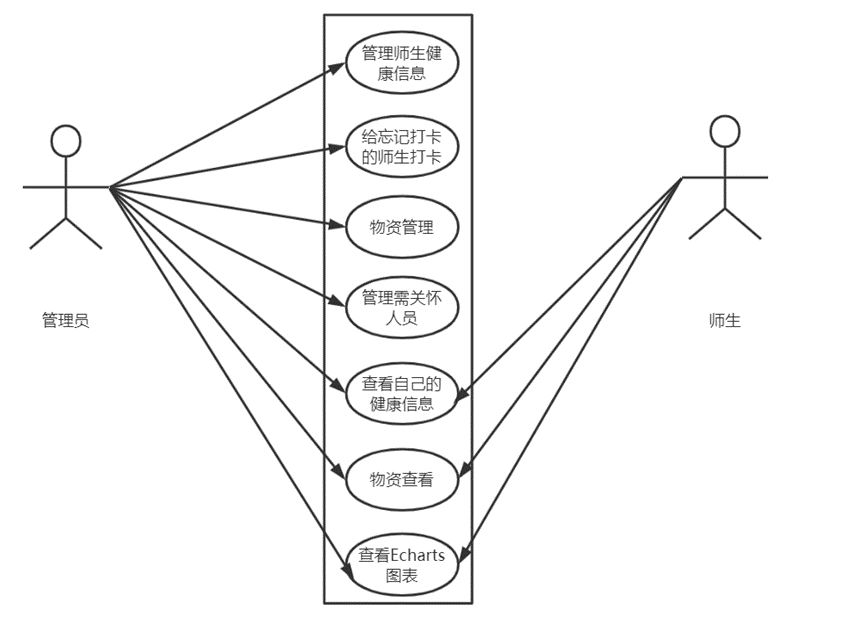
 （5）物资管理模块

    该模块主要用于管理重要的卫生物资，有效地应对突发的校园疫情。

图示, 示意图

描述已自动生成

**图7 后端功能模块图**



**图8 后台系统用例图**

## 数据库设计

本系统采用Mysql作为非关系型数据库，采用innodb存储引擎和可重复读的隔离级别，有效防止多并发场景下的脏读不可重复读等问题。MySQL还具有良好的扩展性，能够轻松地进行水平和垂直扩展[11]。水平扩展可以通过添加更多的服务器来实现，而垂直扩展可以通过增加单个服务器的处理能力来实现。该系统主要有以下表以下数据库表：用户表（user）,学生记录表（student），教师记录表（teacher），卫生物资表（material），需关怀人员表（important），院系表（depart)。

### 数据库实体模型设计

总的ER图如下图9所示，教师、学生和需关怀人员都属于院系，是n对n的关系。教师、学生可以在健康记录表上面打卡，是n对n的关系。管理员可以对健康记录表进行管理，是1对n的关系，也可以对卫生物资管理，也是1对n的关系。

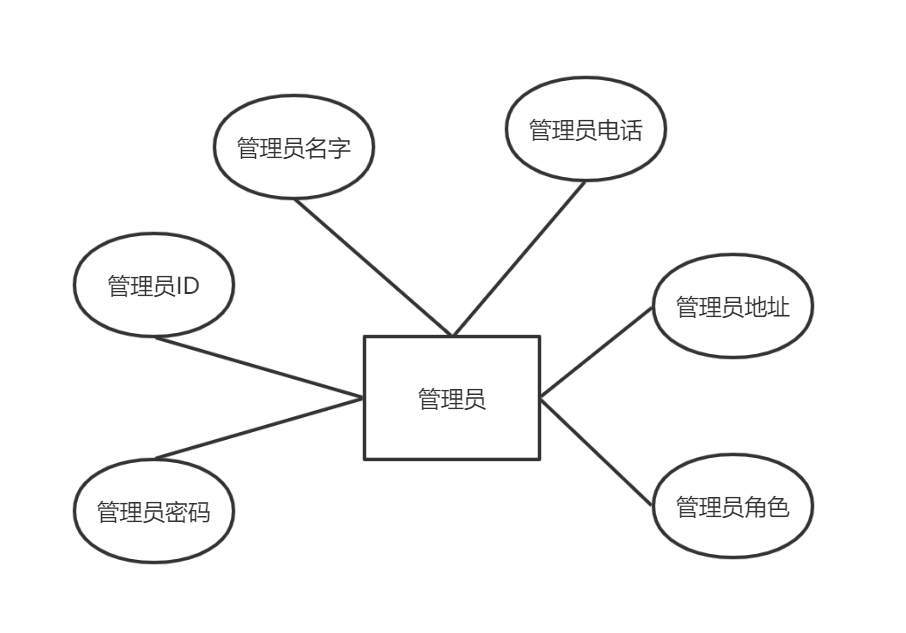
图示

描述已自动生成

**图9 ER图**

1. 管理员集合

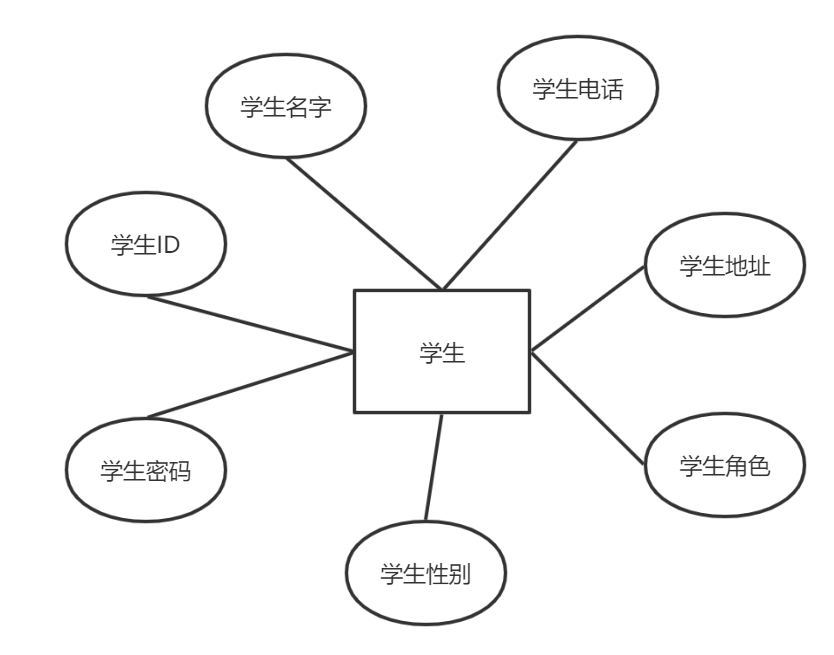
管理员主要负责对师生的健康打卡信息进行管理，对于失误的上传健康信息给予及时的修改，同时对需关怀人员进行定期的更新，对Echarts的图表数据进行及时的更新，对卫生物资进行相关的管理等，如图10所示。



**图10 管理员实体图**

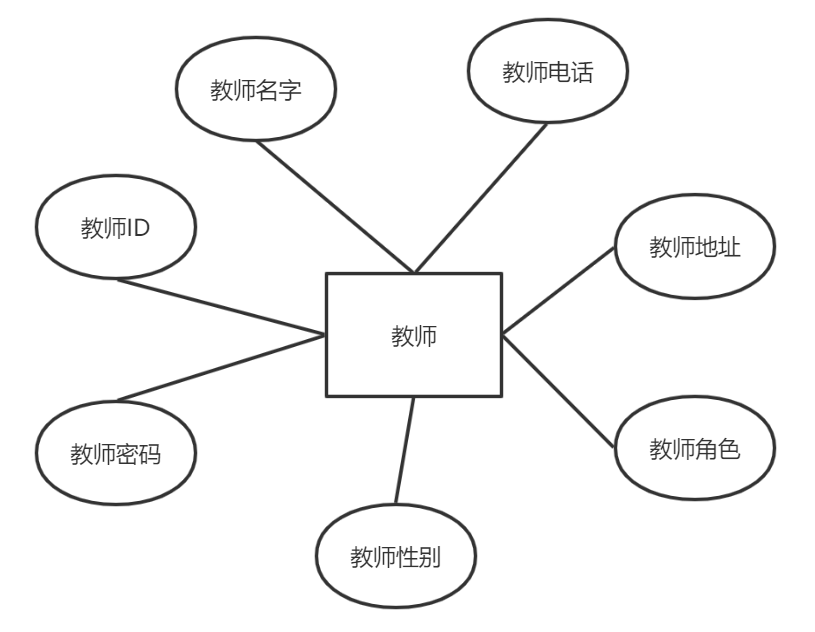
1. 学生集合

学生主要可以上传自己的个人健康信息，同时查看前端的图表，直观地体验流感的变化等，加强自己的卫生意识。如图11所示。



**图11 学生实体图**

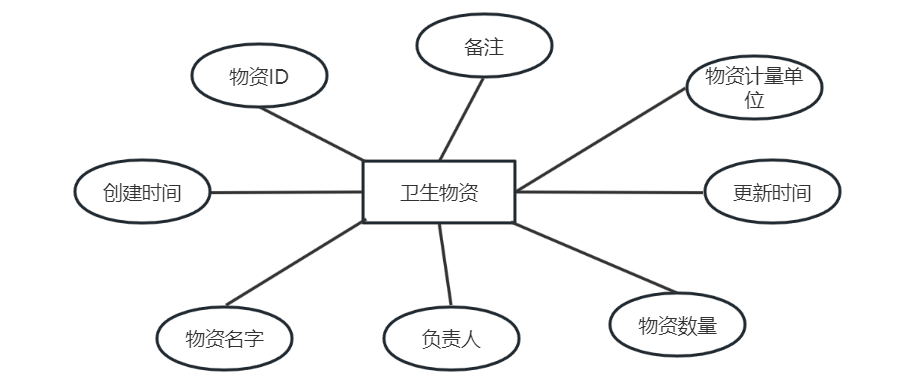
1. 教师集合

教师主要可以上传自己的个人健康信息，同时查看前端的图表，直观地体验流感的变化等，加强自己的卫生意识，也可以对学生的健康信息进行一定的管理，方便整个流程的执行，如图12所示。

**图12 教师实体图**

1. 卫生物资集合

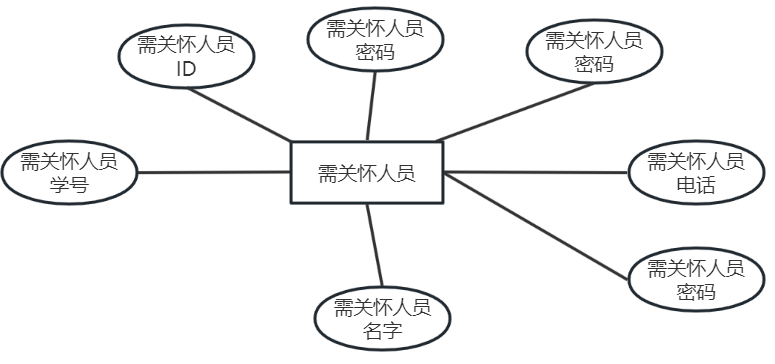
主要对卫生物资进行一些增删改查操作，方便其他Echarts表格获取数据和对卫生物资汇总，如图13所示。



**图13 卫生物资实体图**

1. 关怀人员集合

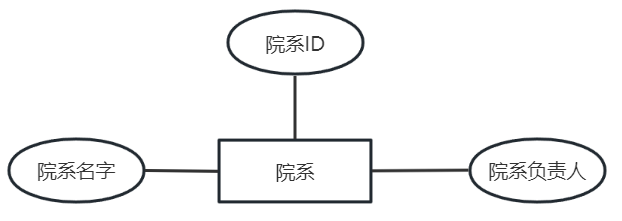
主要是针对一些感冒发烧等流感症状比较严重的人群，各级学院的负责人可以查看这里的人群，给予一定的关心和帮助。如图14所示。



**图14 需关怀人群实体图**

（6）院系集合

主要负责对院系进行管理，对负责人信息等的及时更新，如图15所示。



**图15 院系实体图**

### 数据库结构设计

数据库结构设计描述的是数据的组织和存储，下面是根据上一节给出的实体-关系图，将其转化成符合数据库规范的结构设计，结合实际做出主要数据表的结构设计。

（1）用户表主要是存储用户的一些基本信息，打卡的时候可以直接默认填充。可以自己根据该表修改个人基本信息，用户的密码可以自己修改，也可以由管理员后台修改，角色信息Role只能由管理员进行分配和修改。如表1所示。

表 1 用户表

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **说明** |
| Id | Int | 用户ID |
| UserId | String | 用户学号 |
| Depart | String | 用户院系 |
| Nickname | String | 用户名字 |
| Password | String | 用户密码 |
| Role | Int | 用户角色 |
| Sex | Int | 性别 |

（2）学生记录表主要是记录学生的健康信息，由学生的基本信息如学号、名字、地址、院系等组成，也有个人打卡上传的健康情况和体温等信息。如表2所示。

表 2 学生记录表

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **说明** |
| Id | Int | 主键 |
| StuId | String | 学生学号 |
| Nickname | String | 学生名字 |
| Weizhi | String | 学生地址 |
| Phonenum | String | 学生电话 |
| Password | String | 学生密码 |
| Sex | String | 学生性别 |
| Depart | String | 所属院系 |
| CreateTime | Datetime | 创建时间 |
| Temp | Float | 当前体温 |
| Health | Varchar | 是否健康 |
| Role | Int | 学生角色 |
| Content | String | 备注 |

（3）卫生物资表主要负责记录学校库存的卫生物资情况，主要有物资名字、物资数量、物资计量单位、负责人个人信息组成。创建时间和更新时间主要用于记录物资修改的时间，方便Echarts表格根据日期动态地汇总展示。如表3所示。

表 3 卫生物资表

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **说明** |
| Id | Int | 物资id |
| Name | String | 物资名字 |
| Count | Integer | 物资数量 |
| Type | String | 物资计量单位 |
| Charge | String | 负责人 |
| Cnum | String | 负责人电话 |
| CreateTime | Datetime | 创建时间 |
| UpdateTime | Datetime | 更新时间 |
| Content | String | 物资备注 |

（4）需关怀人员记录表主要是记录需关怀人员的健康信息，由需关怀人员的基本信息如学号、名字、地址、院系、性别、电话等组成，也有个人打卡上传的健康情况和体温等信息，还新增了个确诊时间字段，方便管理员和各院系的教师及时管控当天的重点人员。如表4所示。

表4 需关怀人员记录表

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **说明** |
| Id | Int | 主键 |
| UserId | String | 需关怀人员学号 |
| Nickname | String | 需关怀人员名字 |
| Weizhi | String | 需关怀人员地址 |
| Phonenum | String | 需关怀人员电话 |
| Sex | String | 需关怀人员性别 |
| Depart | String | 需关怀人员院系 |
| CreateTime | DateTime | 创建时间 |
| Temp | Float | 当前体温 |
| Content | String | 备注 |
| Idate | DateTime | 确诊时间 |

（5）院系表主要是记录院系的基本信息，由院系名字、院系负责人、负责人电话、更新时间和创建时间组成，有异常情况可通过电话找相关负责人。如表5所示。

表 5 院系表

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **说明** |
| Id | Int | 院系id |
| Name | String | 院系名字 |
| Charge | String | 院系负责人 |
| PhoneNum | String | 负责人电话 |
| CreateTime | Datetime | 创建时间 |
| UpdateTime | DateTime | 更新时间 |

（6）角色表主要记录各用户之间的角色信息，有管理员、学生和教师三种角色。如表6所示。

表 6 角色表

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **说明** |
| Id | Int | 主键id |
| Name | String | 角色名字 |
| Description | String | 角色描述 |
| CreateTime | Datetime | 创建时间 |

（7）教师记录表主要是记录教师的健康信息，由教师的基本信息如学号、名字、地址、院系、性别、电话等组成，也有个人打卡上传的如健康情况和体温等信息。如表7所示。

表 7 教师记录表表

|  |  |  |
| --- | --- | --- |
| **字段名** | **数据类型** | **说明** |
| Id | Int | 主键 |
| TeaId | String | 教师学号 |
| Nickname | String | 教师名字 |
| Weizhi | String | 教师地址 |
| Phonenum | String | 教师电话 |
| Password | String | 教师密码 |
| Sex | String | 教师性别 |
| Depart | String | 所属院系 |
| CreateTime | Datetime | 创建时间 |
| Temp | Float | 当前体温 |
| Health | Varchar | 是否健康 |
| Role | Int | 教师角色 |
| Content | String | 备注 |

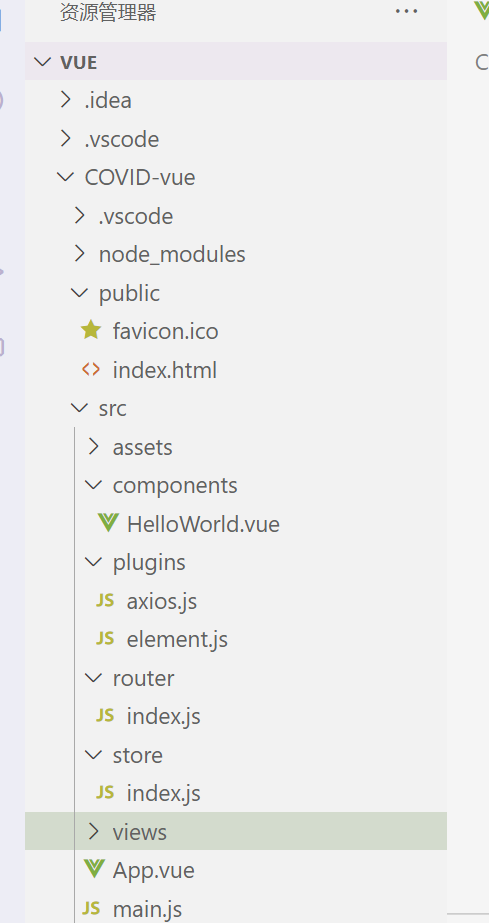
# 系统详细设计与实现

## 系统界面设计

### 前台界面

本项目使用VsCode开发小程序端，项目包含几个主要部分，分别是node\_modules，public、assets、components、plugins、router、store和views，如图16所示。

其中node\_modules在Vue.js项目中，开发人员通常使用它存储vue的依赖；public这个目录包含了所有不需要被打包的静态资源，如图片和图标等；assets这个目录包含了如样式表、图片、字体等被vue使用的静态资源，component:这个目录包含了所有Vue.js组件。组件是Vue.js应用程序的基本单元，每个组件都包含了自己的模板、逻辑和样式；plugins这个目录包含了所有Vue插件，可以通过Vue.use()方法在Vue.js应用程序中安装和使用；router这个目录包含了Vue.js的路由器配置。路由器是Vue.js的官方路由管理工具，可以用来管理应用程序的页面导航；store这个目录包含了Vue.js的全局变量状态管理。views这个目录包含了所有的Vue.js视图[12]。视图是指应用程序中呈现给用户的页面，通常包含了多个组件和模板。



**图16 前台项目结构示意图**

登录界面主要用于用户登录，如图17所示。

图形用户界面, 应用程序, 网站

描述已自动生成

**图17 登录界面示意图**

登录界面非空校验。如果输入空的用户名或密码或进行校验，并提示不能为空，如图18所示。

文本

描述已自动生成

**图18 登录界面非空校验示意图**

注册界面主要用于用户注册，如图19所示。



**图19 注册界面示意图**

前台展示页面主要用于展示师生们的健康情况，如图20所示。



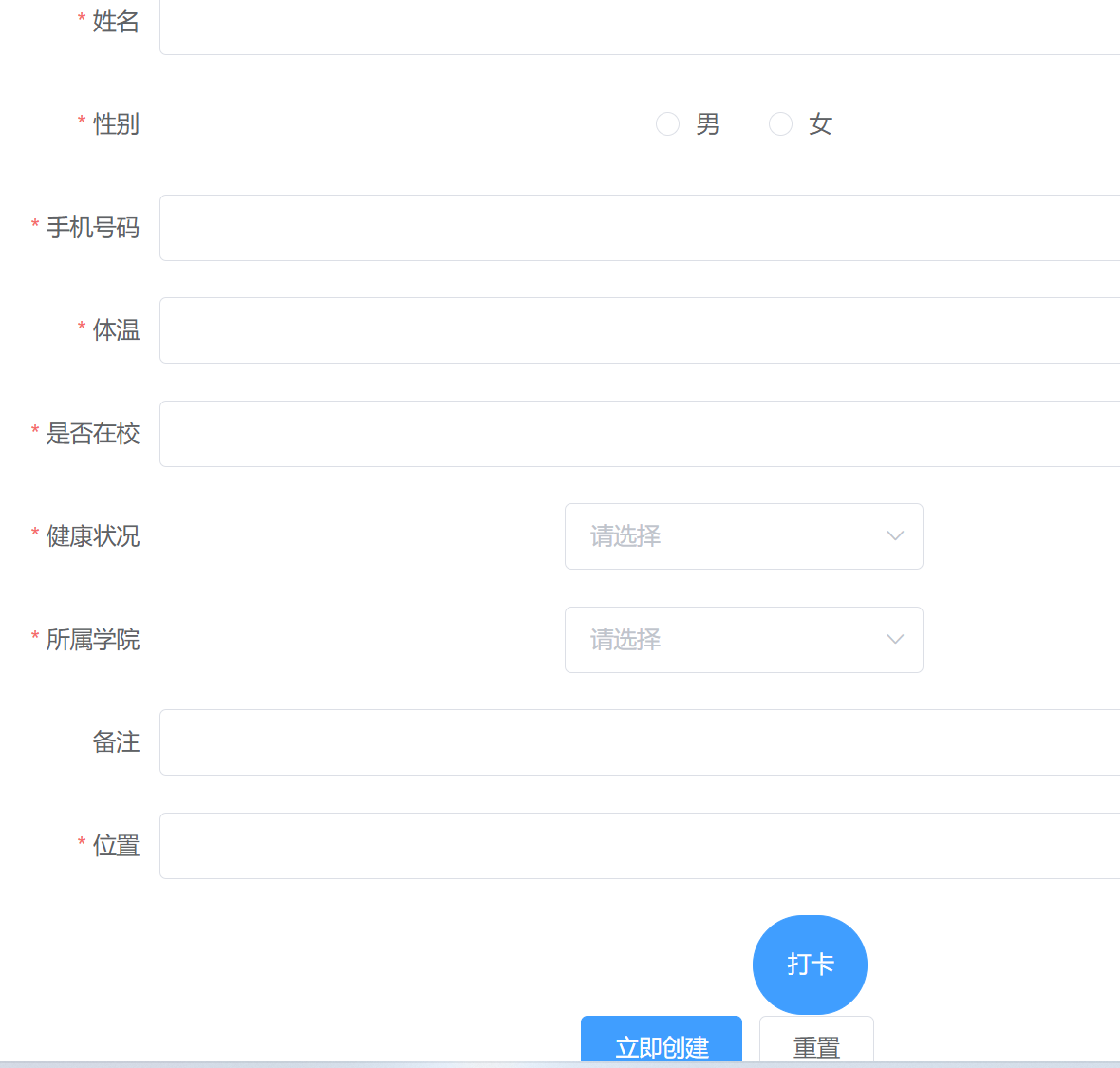
**图20 首页展示示意图**

个人信息页面主要用于展示师生们的个人信息，查看是否有误，如图21所示。

****

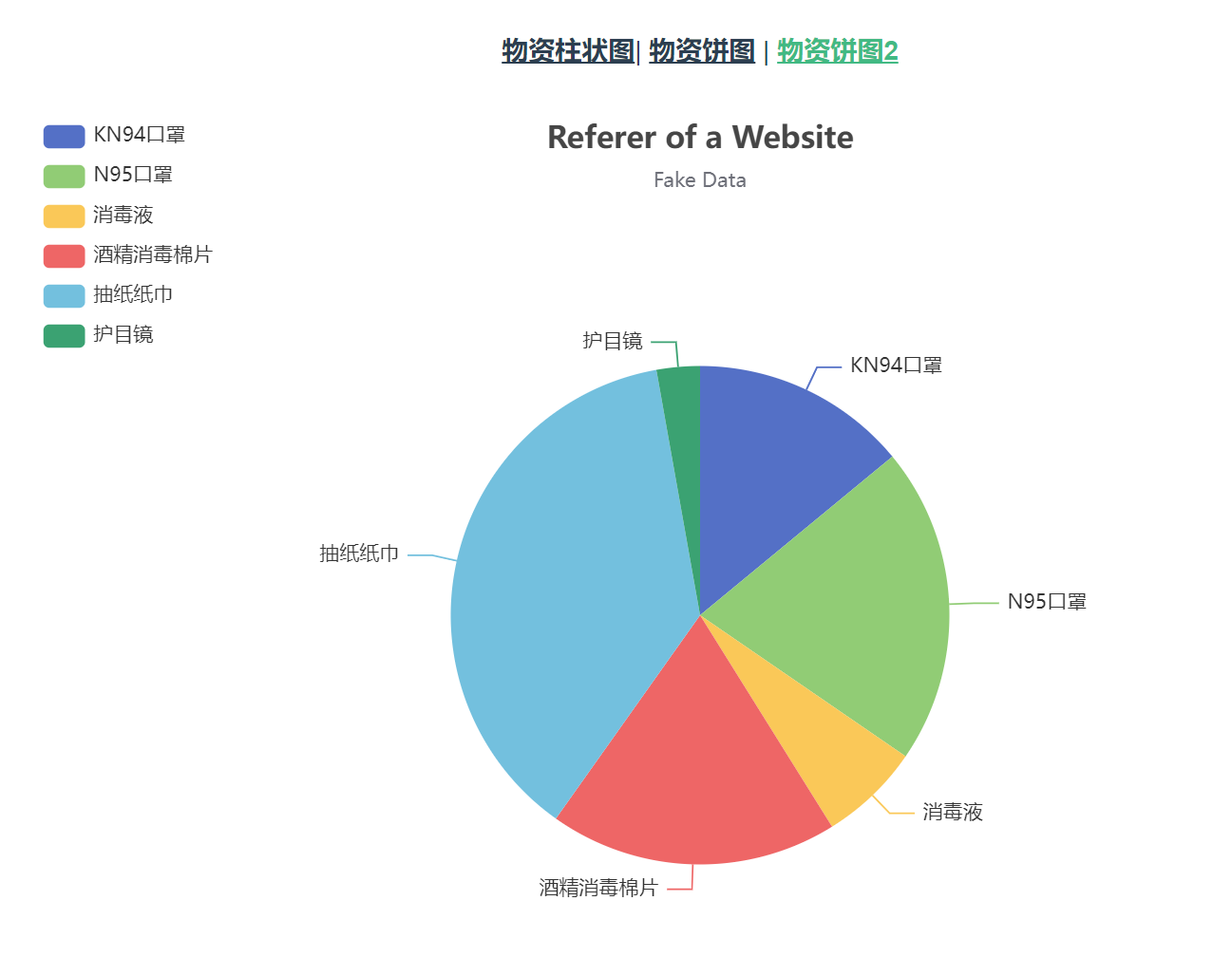
**图21 个人信息示意图**

打卡页面主要用于师生们的每日健康打卡，上传个人健康信息，如图22所示。

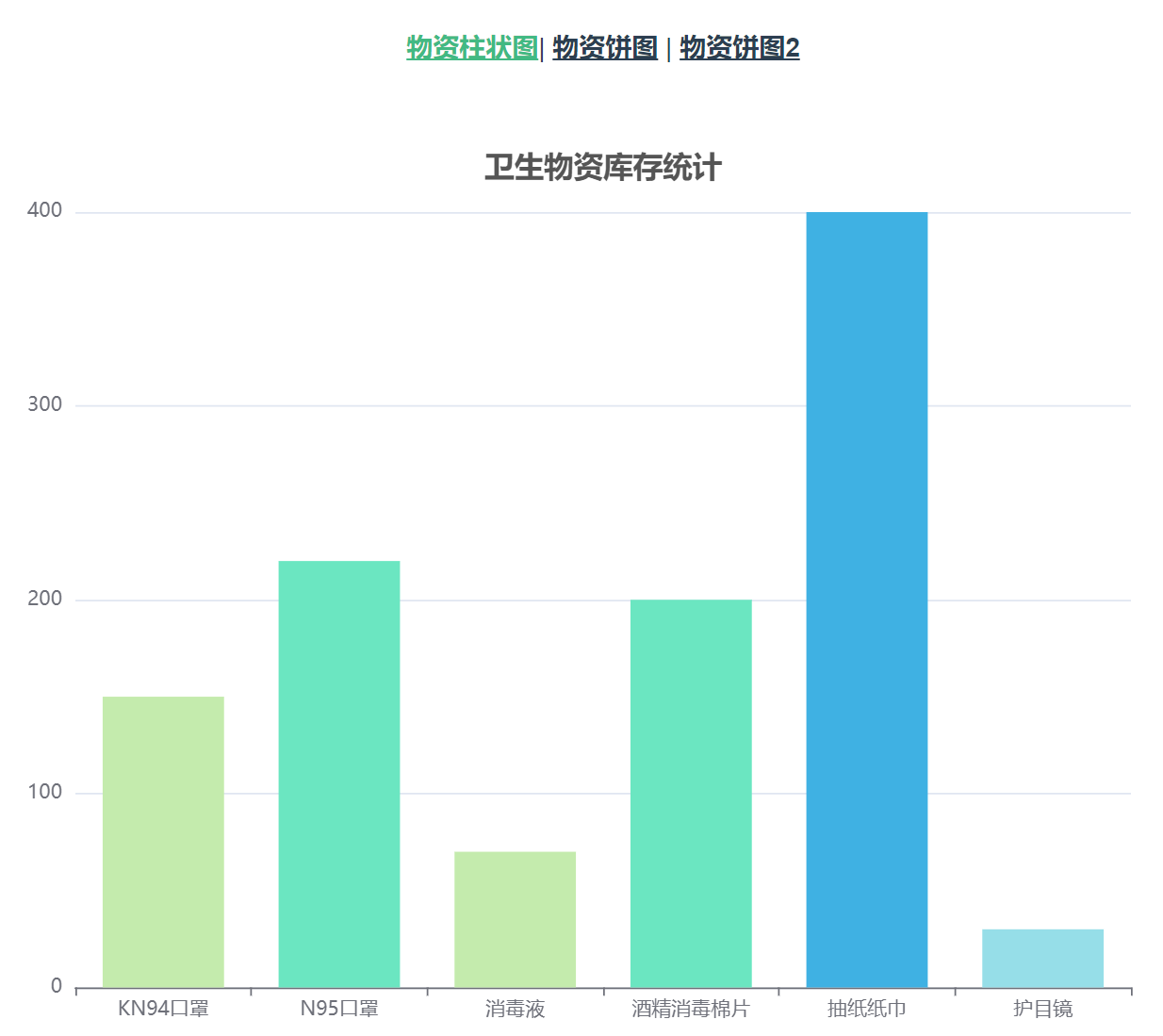
****

**图22 打卡功能示意图**

物资展示页面主要用于用柱状图、饼状图展示物资情况，如图23、图24所示。



**图23 物资展示饼图**

****

**图24 物资展示柱状图**

### 5.1.2后台界面

主要后台模块，如图25所示。日常防控管理，管理员可以帮助忘记打卡的师生进行打卡，然后汇总师生的打卡信息。人员管理主要负责对用户进行增删改查，还有用户权限的管理。管理员可以赋予其他用户管理员、教师和学生三种权限。教师健康管理主要是对提交上来的教师打卡信息进行管理。学生健康管理也是对学生提交上来的打卡信息进行管理。关怀人员管理是对确诊流感的师生们进行统一的管理，方便图表数据的采集。卫生物资管理是解决纸质资料易丢失的问题，对学校的卫生物资进行统一的管理。流感情况可视化主要是对于确诊人员近几个月的折线图展示和卫生物资的Echarts图表展示。

文本

描述已自动生成

**图25 后台界面模块功能示意图**

日常防控管理，主要对一些未及时打卡同学的健康信息进行上传，如图26所示。



**图26 健康信息上传示意图**

需关怀人员登记，主要负责对一些症状严重，影响日常工作学习的师生健康状况进行记录观察，必要时给予帮助，如图27所示。

图形用户界面, 应用程序

描述已自动生成

**图27 需关怀人员登记示意图**

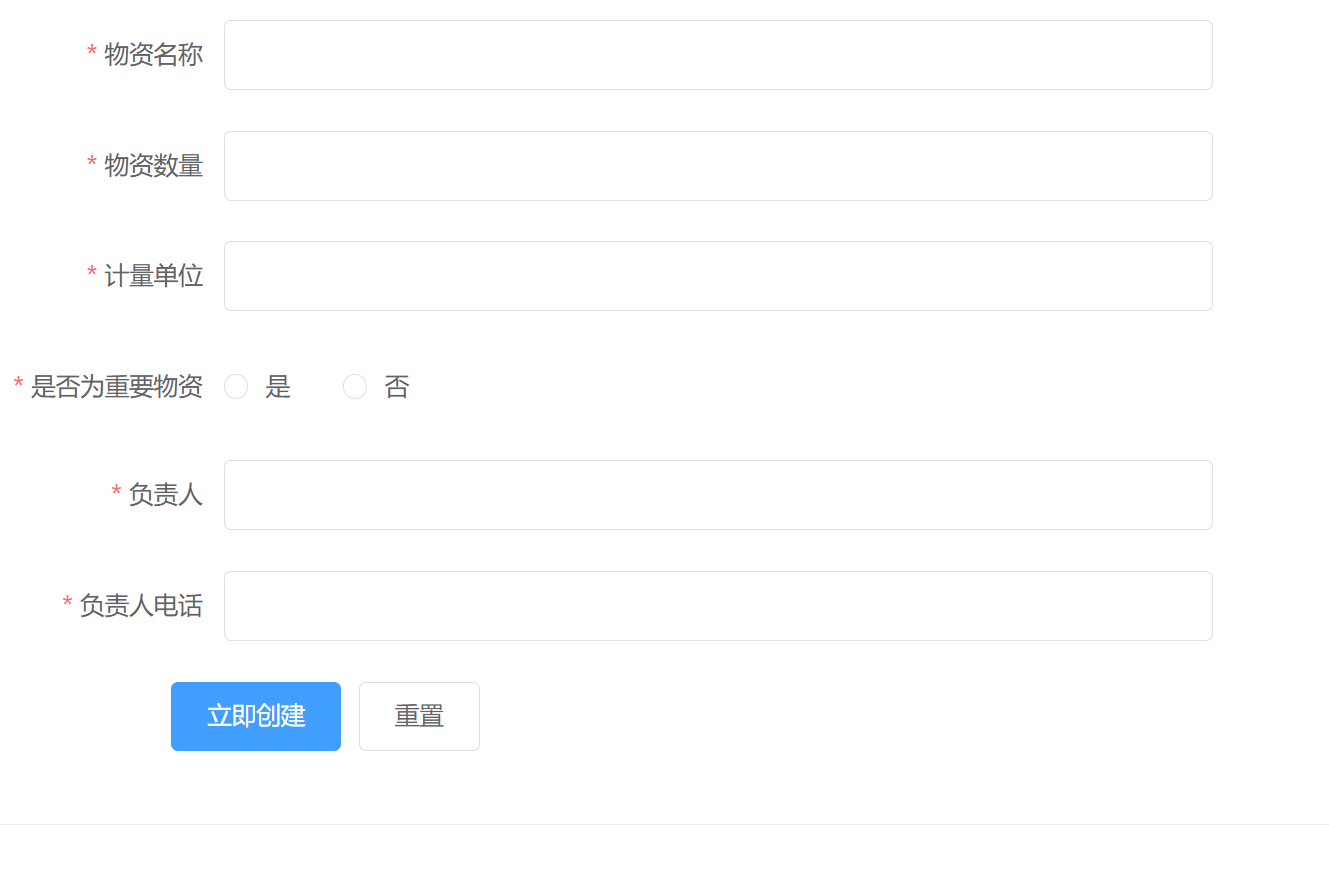
对需关怀人员进行修改，如图28所示。

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

**图28 需关怀人员修改功能示意图**

卫生物资管理用于管理卫生物资的模块，它可以帮助学校有效地管理和调配医疗用品和药品等物资，跟踪物资的库存情况，科学地分配物资，确保物资的充足供应和合理使用。如图29所示。



**图29 卫生物资管理示意图**

卫生物资展示，主要对卫生物资进行一些简单的增删改查操作，如图30所示。

表格

描述已自动生成

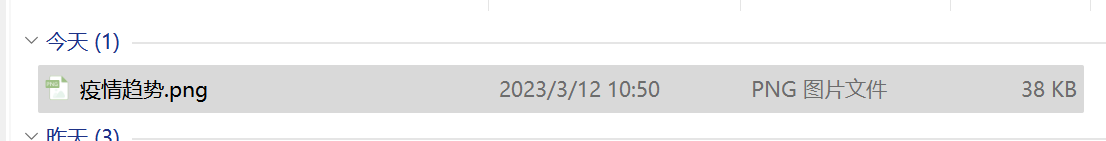
**图30卫生物资管理示意图**

Echarts展示模块:通过对流感数据的可视化展示，学校领导师生可以更加全面地了解师生们情况，及时采取应对措施，调整卫生预防策略，提高决策效率和应对能力，如图31所示。



**图31 Echarts展示模块示意图**

提供了下载功能可以在电脑文件中打开。如图32所示。



**图32下载功能示意图**

下载后的图片可以打开，如图33所示。

图表, 折线图

描述已自动生成

**图33 下载文件打开示意图**

此外也有大量正则表达式校验，如图34所示。

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

**图34 打卡功能非空校验示意图**

疫情防控管理系统部署界面，如图35所示。

图形用户界面, 文本, 应用程序

描述已自动生成

**图35 系统部署示意图示意图**

## 后台服务端

本次毕设项目采用前后端分离架构，其中前端在HTTP请求头Header中携带session，后端再根据前端的请求处理对应逻辑，通过MybatisPlus来调用数据库来返回相关的数据，有效减少繁琐冗长的配置。此外还用了JDBC编写复杂Sql实现统计功能。

这次的系统开发过程中遇到过许多问题，在使用Echarts表格的时候，经常会显示不出数据，或者图表数据出现如位置错乱，图表位置不对等情况，这主要是因为对List、ArrayList、Map等数据类型使用不熟练导致，对于一些前台界面的集合数据，后台的java代码因为粗心等原因导致整个表格出现问题。此外日期时间的设置问题，用错了String类型来编写，导致始终不能够自动地插入创建时间和更新时间，同时api也不太熟悉，使得开发的时候花了不少时间调试bug。

调试的时候也存在不小的问题，前后端对接的时候经常报CROS跨域问题，通过查阅资料和询问开发经验丰富的同事才知道触发了浏览器的跨域机制，这是浏览器中的一种安全策略，为了限制协议、域名、端口不相同之间的交互，避免一些浏览器之间的攻击，不同源的请求是无法进行交互来保证安全。为了解决这些问题，我寻找了很多解决方案，其中不管是添加一个CROS过滤器还是重写addCrosMappings方法都很有用，我也因此接触到不少Spring开源框架不少的底层源码[13]。当然最吸引我的还是用Vue-route的反向代理机制，在相关的配置文件中changeOrigin=false的地方改成True即可解决，在改完这个bug不仅感慨框架作者的技术高超，一个天才的架构师，将底层逻辑封装好，给工程师调用，就能节省不少时间解决繁琐细碎的配置问题，更加专注于业务，把我们从配置陷阱中解放出来，从而创造更多的社会价值。

# 系统测试

本系统测试的是疫情防控管理系统的系统测试用例，主要有以下模块，登录注册模块、师生打卡模块、图表展示模块、物资管理模块、需关怀人员管理模块。主要根据功能模块进行测试，采用黑盒测试方法。

**表8 登录注册模块测试表**

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例** | **预期结果** | **实际结果** | **结论** |
| 用户名错误时点击登录，系统提示“用户名或密码错误” | 弹出提示框，提示用户名或密码错误 | 弹出提示框，提示用户名或密码错误 | 通过测试用例 |
| 用户名和密码出错时点击登录，系统提示“用户名或密码错误” | 弹出提示框，提示用户名或密码错误 | 弹出提示框，提示用户名或密码错误 | 通过测试用例 |
| 用户名和密码正确时点击登录，跳转到主页面 | 进入主页面，提示登录成功 | 进入主页面，提示登录成功 | 通过测试用例 |
| 输入已注册的用户名和密码进行注册，系统提示“该用户已存在” | 弹出提示框，提示用户该用户名已被注册 | 弹出提示框，提示用户该用户名已被注册 | 通过测试用例 |
| 输入未注册用户名和密码注册，系统提示“注册成功” | 弹出提示框，提示用户注册成功 | 弹出提示框，提示用户注册成功 | 通过测试用例 |
| 注册时用户名或者密码为空，提示“用户名或者密码不能为空” | 弹出提示框，提示用户名或者密码不能为空 | 弹出提示框，提示用户名或者密码不能为空 | 通过测试用例 |

**表9 师生打卡模块测试表**

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例** | **预期结果** | **实际结果** | **结论** |
| 点击打卡按钮，如果该用户已打卡，则打卡按钮不可用 | 点击打卡按钮，如果该用户已打卡，则打卡按钮不可用 | 点击打卡按钮，如果该用户已打卡，则打卡按钮不可用 | 通过测试用例 |

**表10 图表展示模块测试表**

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例** | **预期结果** | **实际结果** | **结论** |
| 多图表展示各学院的健康状况 | 看到展示各学院的健康状况的图表 | 看到展示各学院的健康状况的图表 | 通过测试用例 |

**表11 物资管理模块测试表**

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例** | **预期结果** | **实际结果** | **结论** |
| 点击添加物资，输入物资名称，提示添加成功 | 弹出提示框，提示添加成功 | 添加成功 通过测试用例 | 通过测试用例 |

**表12 需关怀人员管理模块测试表**

|  |  |  |  |
| --- | --- | --- | --- |
| **测试用例** | **预期结果** | **实际结果** | **结论** |
| 点击添加需关怀人员管理，输入人员信息，提示添加成功 | 弹出提示框，提示添加成功 | 添加成功 通过测试用例 | 通过测试用例 |

# 结论

## 总结

本次毕业设计我做的是疫情防控管理系统，利用百度地图等大公司的位置服务API，完成了登录注册模块，师生打卡模块，图表展示模块，物资管理模块，需关怀人员管理模块，这其中的开发过程肯定是艰难曲折的，因为面对的不仅是论文毕设等的压力，还有实习和就业方面的压力，时间上比较紧迫，有缺陷之处还望各位专家、老师和同学们海涵。当然这其中我收获也良多，从查阅资料确定选题，需求分析，概要设计和详细的数据库设计，再到功能模块的划分，到用自己学会的Springboot和Vue技术去实现，最后整个系统完成的那一刻，内心的喜悦是按捺不住的。整个过程中我也遇到不少的困难，有技术上的，有经验上的，也有生活上的，但都在老师、朋友、同事的帮助下一一战胜，非常感谢他们对我的无私帮助。

## 展望

目前的系统功能有限，而且平台也只能限于网站，容易被抓包受到DDOS攻击而宕机，需要更先进的网络安全技术来丰富系统，由于个人的时间和技术水平有限未能实现，还有就是希望能够新增通过海量的数据，通过一定的算法模型来进行未来的流感趋势等功能，由于本人对Python人工智能、神经网络等方面的研究不够深入，致使该部分功能只能放在未来展望中，在今后的职业生涯中，我会认真工作积极学习先进技术，让自己能够实现这一功能，更好地完善我这个系统，从而更好地为保障人民健康和社会安定的社会主义卫生事业做出微薄贡献。

**参考文献**

1. 郭洪涛,郑光,赵静,姜淼,何晓娟,吕爱平. 基于文本挖掘分析甲型H1N1流感的中医药治疗特色[J].世界科学技术,2011,13(05):772-776.
2. 梁弼,许晓红,周思吉. 基于SSH的高校教师教学质量评价系统的研究与实现[J].软件导刊,2011,10(02):91-93.
3. 张卫辉,赵彦云. 突发重大传染病疫情数据互联网统计体系研究——以政府数据开放平台新冠肺炎疫情数据开放为例[J].统计研究,2022,39(05):49-62.
4. 孙存志,杨丽. “一带一路”沿线卫生健康共同体构建研究[J].青海社会科学,2020(05):70-75.
5. 陈劳,刘金扬,杜昀徽,丁逸. 高校新冠疫情防控系统模式设计与开发[J].中国教育信息化,2021(05):76-79.
6. 万燕,张博锐. 基于前后端分离的防伪溯源系统[J].智能计算机与应用,2020,10(07):82-86.
7. 程斌斌,程昊南,朱洪军. 基于Spring Boot技术的重点人员监测系统的设计与实现[J].电脑知识与技术,2021,17(03):103-105.
8. 李昊,马雪,李晓磊. 基于HTML5的潮白河补水APP关键技术研究与应用[J].水资源开发与管理,2022,8(04):75-79.
9. 朱锋. 基于SpringBoot和Vue的火电SIS系统设计与实现[J].科技创新导报,2019,16(26):36-37.
10. 虞猷. 基于Android的智能课堂设计与实现[D].江西财经大学,2016.
11. 宋涛,李秀华,李辉,文俊浩,熊庆宇,陈杰. 大数据时代下车联网安全加密认证技术研究综述[J].计算机科学,2022,49(04):340-353.
12. Franklin J,Wanyoike M,Bouchefra A. Working with Vue.js[M]. SitePoint,2019.
13. Cheng F. Talent Recruitment Management System for Small and Micro Enterprises Based on Springboot Framework[J].Advances in Educational Technology and Psychology,2021,5(2):99-105.

**致 谢**

通过本次管理系统的开发，我学习到不少先进的主流技术框架，我的语言组织能力与课题研究能力也有所提升。此次毕业设计工作从开题报告到论文基本顺利实现，离不开我的指导老师熊建芳导师的悉心指导，她给我指点了大致的方向，让我受益匪浅。

同时，我还要感谢我工作的同事们，他们包容和支持让我能够很好的开展工作，他们在解决BUG等方面给予了我帮助，让我能够更好地完成我的研究工作。

此外，我还要感谢我的家人和朋友，他们一直以来的支持和鼓励让我克服了许多挑战，并且在我遇到困难时给予了我无私的帮助和鼓励，让我战胜困难走了过来，

在此，再次向所有对我的论文工作给予的人们表示深深的感谢！