**后疫情时代高校宿舍管理系统小程序**

# **摘 要**

随着科学技术的飞速发展，社会的方方面面、各行各业都在努力与现代的先进技术接轨，通过科技手段来提高自身的优势，高校当然也不例外。后疫情时代高校宿舍管理系统小程序是以实际运用为开发背景，运用软件工程原理和开发方法，采用Java技术构建的一个管理系统。整个开发过程首先对软件系统进行需求分析，得出系统的主要功能。接着对系统进行总体设计和详细设计。总体设计主要包括系统功能设计、系统总体结构设计、系统数据结构设计和系统安全设计等；详细设计主要包括系统数据库访问的实现，主要功能模块的具体实现，模块实现关键代码等。最后对系统进行功能测试，并对测试结果进行分析总结，得出系统中存在的不足及需要改进的地方，为以后的系统维护提供了方便，同时也为今后开发类似系统提供了借鉴和帮助。

本系统使用Java技术开发，SpringBoot框架，采用的数据库是Mysql。在设计过程中，充分保证了系统代码的良好可读性、实用性、易扩展性、通用性、便于后期维护、操作方便以及页面简洁等特点。

关键词：后疫情时代高校宿舍管理系统小程序，Java，SpringBoot，Mysql

Post epidemic era college dormitory management system mini program

ABSTRACT

With the rapid development of science and technology, all aspects of society and industries are striving to integrate with modern advanced technology and improve their own advantages through technological means, and universities are no exception. The post epidemic era college dormitory management system mini program is a management system built on the background of practical application, using software engineering principles and development methods, and Java technology. The entire development process starts with a requirement analysis of the software system to determine its main functions. Then proceed with the overall and detailed design of the system. The overall design mainly includes system functional design, system overall structure design, system data structure design, and system security design; The detailed design mainly includes the implementation of system database access, the specific implementation of main functional modules, and the key code for module implementation. Finally, functional testing was conducted on the system, and the test results were analyzed and summarized to identify the shortcomings and areas for improvement in the system. This provided convenience for future system maintenance and also provided reference and assistance for the development of similar systems in the future.

This system is developed using Java technology and the SpringBoot framework, using MySQL as the database. During the design process, the system code was fully guaranteed to have good readability, practicality, scalability, universality, ease of later maintenance, easy operation, and concise pages.maintenance, easy operation, and concise pages.

**Key words:** post epidemic era university dormitory management system mini program, Java, SpringBoot, MySQL

目 录

[摘 要 I](#_Toc12493)

[第1章 绪论 1](#_Toc21002)

[1.1背景及意义 1](#_Toc24521)

[1.2 国内外研究概况 1](#_Toc31636)

[1.3 研究的内容 2](#_Toc19656)

[第2章 相关技术 3](#_Toc20574)

[2.1本UNI-APP小程序框架 3](#_Toc28981)

[2.2 MySQL数据库 3](#_Toc6232)

[2.3 B/S结构 3](#_Toc8361)

[2.4 Springboot框架 4](#_Toc32372)

[2.5 Java简介 4](#_Toc28937)

[2.6 MVVM模式 5](#_Toc25820)

[第3章 系统分析 6](#_Toc6240)

[3.1 需求分析 6](#_Toc15306)

[3.2 系统可行性分析 6](#_Toc5850)

[3.2.1技术可行性 6](#_Toc30768)

[3.2.2经济可行性 7](#_Toc30027)

[3.2.3操作可行性 7](#_Toc28819)

[3.3 项目设计目标与原则 7](#_Toc25887)

[3.4系统流程分析 8](#_Toc22980)

[3.4.1操作流程 8](#_Toc13484)

[3.4.2添加信息流程 9](#_Toc10844)

[3.4.3删除信息流程 10](#_Toc25408)

[第4章 系统设计 11](#_Toc9002)

[4.1 系统体系结构 11](#_Toc14938)

[4.2系统结构设计 11](#_Toc27631)

[4.3开发流程设计 11](#_Toc15856)

[4.4 数据库设计原则 12](#_Toc28445)

[4.5 数据表 13](#_Toc5047)

[第5章 系统详细设计 24](#_Toc13432)

[5.1管理员功能模块 24](#_Toc16829)

[5.2学生用户功能模块 29](#_Toc14787)

[5.3辅导员功能模块 36](#_Toc26337)

[5.4宿管功能模块 41](#_Toc486)

[第6章 系统测试 44](#_Toc25573)

[6.1系统测试的目的 44](#_Toc5679)

[6.2系统测试方法 44](#_Toc25133)

[6.3功能测试 45](#_Toc21140)

[结 论 46](#_Toc19820)

[致 谢 47](#_Toc23850)

[参考文献 48](#_Toc16337)

# **第1章 绪论**

## 1.1背景及意义

在高校里面，自新冠疫情爆发以来，高校宿舍管理成了高校的重点工作之一，高校宿舍管理是高校工作信息化建设中的重要环节。进入后疫情时代，疫情防控信息管理越来越正规化、科学化，也势必要引入系统对这些信息进行管理。因此开发一个方便快捷易于使用的后疫情时代高校宿舍管理系统是很有必要的，它将成为今后疫情防控不可缺少的一部分。在日常使用过程中，系统提供的方便，使得工作效率得以快速提升，并使高校宿舍管理人力资源得以更有效配置，从而提高高校宿舍管理管理水平，也能为是师生提供更安全、安心的环境。

本系统从高校的实际需要出发，以提高工作效率、节约时间为目标，设计开发了一个基于Springboot的后疫情时代高校宿舍管理系统，后疫情时代高校宿舍管理系统围绕系统管理员、学生用户、宿管用户、辅导员四种角色，主要包括后台首页、管理员（个人资料、修改密码）、轮播图（轮播图管理）、资源管理（校园资讯、资讯分类）、系统用户（管理员、学生用户、宿管用户、辅导员）、模块管理（入住申请、请假申请、报修申请、健康上报、晚归申请、留言反馈、通知公告）等功能模块。论文首先介绍了本系统的相关技术，并进行了需求分析，分别给出了系统Web服务器端和移动终端的概要设计、数据库设计和详细设计，实现完成的系统经测试上线，运行效果良好。

本系统主要是借助计算机，对轮播图、资源管理、系统用户等进行管理等。该系统通过和数据库管理系统软件协作来满足用户的需求，能够有效的解决获取信息便捷化、全面化的问题，提高效率。

## 1.2 国内外研究概况

目前国内外传统的高校宿舍管理方式是基于手工的，而人的精力和工作准确度是有限的，所以传统的高校宿舍管理方式难免会存在效率低下、准确率低等缺点，还伴随着人力资源的大量浪费。在管理的工作中，所涉及到的工作电脑机器比人类更为擅长，能够达到快速、准确、耗费资源小等要求。基于这些显著的优点，如果能设计一套完整且贴合高校需求的高校宿舍管理系统，那么将大大地提高高校宿舍管理的工作效率。在具备这样的高校宿舍管理系统之下，高校工作人员只需配备少量的专业人员对数据库进行日常数据审核、管理以及定期维护。整个过程中管理人员只需提供少量的必要信息，大部分数据处理工作皆由数据库和相关程序来完成，节省了大量时间。如果数据库中的某部分信息存在错误，那么管理人员可以通过批量修改等方式来降低排错过程中所要花费的时间和精力。因此，开发一个便于使用的后疫情时代高校宿舍管理系统是十分有必要的。

## 1.3 研究的内容

目前许多人仍将传统的单机设备作为信息管理的主要工具，而网络技术的应用只是起到辅助作用。在对网络工具的认知程度上，较为传统的office软件等仍是人们使用的主要工具，而面向各个专门行业的管理系统的信息管理软件仍然缺乏群众基础。本选题则旨在通过标签分类管理等方式，实现后台首页、管理员（个人资料、修改密码）、轮播图（轮播图管理）、资源管理（校园资讯、资讯分类）、系统用户（管理员、学生用户、宿管用户、辅导员）、模块管理（入住申请、请假申请、报修申请、健康上报、晚归申请、留言反馈、通知公告）等功能。

# **第2章 相关技术**

## 2.1本UNI-APP小程序框架

Uni-app 是一个使用 [Vue.js](https://vuejs.org/" \t "https://uniapp.dcloud.net.cn/_blank) 开发所有前端应用的框架，开发者编写一套代码，可发布到iOS、Android、Web（响应式）、以及各种小程序（微信/支付宝/百度/头条/飞书/QQ/快手/钉钉/淘宝）、快应用等多个平台。

DCloud公司拥有900万开发者、数百万应用、12亿手机端月活用户、数千款Uni-app插件、70+微信/qq群。阿里小程序工具官方内置Uni-app，腾讯课堂官方为Uni-app录制培训课程，开发者可以放心选择。

整个小程序框架系统分为两部分：[逻辑层](https://developers.weixin.qq.com/miniprogram/dev/framework/app-service/)和[视图层](https://developers.weixin.qq.com/miniprogram/dev/framework/view/)。小程序开发框架的目标是通过尽可能简单、高效的方式让开发者可以在微信中开发具有原生小程序 体验的服务。小程序在视图层与逻辑层间提供了数据传输和事件系统，提供了自己的视图层以及逻辑层框架，让开发者能够专注于数据与逻辑。框架的核心是一个响应的数据绑定系统，可以让数据与视图非常简单地保持同步。在逻辑层做数据修改，在视图层就会做相应的更新。

## 2.2 MySQL数据库

科技的进步，给日常带来许多便利：教室的投影器用到了虚拟成像技术，数码相机用到了光电检测技术，比如超市货物进出库的记录需要一个信息仓库。这个信息仓库就是数据库，而这次的校园车辆预约系统也需要这项技术的支持。

用MySQL这个软件，是因为它能接受多个使用者访问，而且里面存在Archive等。它会先把数据进行分类，然后分别保存在表里，这样的特别操作就会提高数据管理系统自身的速度，让数据库能被灵活运用。MySQL的代码是公开的，而且允许别人二次编译升级。这个特点能够降低使用者的成本，再搭配合适的软件后形成一个良好的网站系统。虽然它有缺点，但是综合各方面来说，它是使用者的主流运用的对象。

## 2.3 B/S结构

基于Java技术开发的B/S架构系统，需要借助Tomcat服务器应用程序进行部署运行[2]。用户访问系统的时候，通过浏览器向应用程序服务器端发起访问请求，服务器端的程序在接到用户请求以后，服务器端应用程序对客户请求做出相应，在调用服务器端的业务逻辑程序完成和数据库端的交互，进一步生成相应的HTML/XML数据，最终把结果反馈给浏览器端用户。

在该系统的开发中，开发模式采用B/S架构技术进行实现，通过部署服务器端应用程序，实现用户通过网站域名或者内网IP地址访问系统，实现系统中数据的动态化呈现和管理，加之页面效果的动态化呈现，不仅提升了页面的表现力，而且管理者可以随时更新系统中的各种信息，充分满足管理者和访问用户之间的信息交互。

## 2.4 Springboot框架

Spring框架是[Java平台](https://baike.baidu.com/item/Java%E5%B9%B3%E5%8F%B0/3793459" \t "https://baike.baidu.com/item/_blank)上的一种开源应用框架，提供具有控制反转特性的容器。尽管Spring框架自身对编程模型没有限制，但其在Java应用中的频繁使用让它备受青睐，以至于后来让它作为[EJB](https://baike.baidu.com/item/EJB/144195" \t "https://baike.baidu.com/item/_blank)（EnterpriseJavaBeans）模型的补充，甚至是替补。Spring框架为开发提供了一系列的解决方案，比如利用控制反转的核心特性，并通过依赖注入实现控制反转来实现管理对象生命周期容器化，利用面向切面编程进行声明式的事务管理，整合多种持久化技术管理数据访问，提供大量优秀的Web框架方便开发等等。Spring框架具有控制反转（IOC）特性，IOC旨在方便项目维护和测试，它提供了一种通过Java的反射机制对Java对象进行统一的配置和管理的方法。Spring框架利用容器管理对象的生命周期，容器可以通过扫描XML文件或类上特定Java注解来配置对象，开发者可以通过依赖查找或依赖注入来获得对象。Spring框架具有[面向切面编程](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%88%87%E9%9D%A2%E7%BC%96%E7%A8%8B/6016335" \t "https://baike.baidu.com/item/_blank)（AOP）框架，SpringAOP框架基于代理模式，同时运行时可配置；AOP框架主要针对模块之间的交叉关注点进行模块化。Spring框架的AOP框架仅提供基本的AOP特性，虽无法与AspectJ框架相比，但通过与AspectJ的集成，也可以满足基本需求。

## 2.5 Java简介

Java主要采用CORBA技术和安全模型，可以在互联网应用的数据保护。它还提供了对EJB（Enterprise JavaBeans）的全面支持，Java servlet API，JSP（Java server pages），和XML技术。Java语言是一种面向对象的语言，它通过提供最基本的方法来完成指定的任务，开发者只需要知道一些概念就能够编写出一些应用程序。Java程序相对较小，其代码能够在小机器上运行。Java是一种计算机编程语言，具有封装、继承和多态性三个主要特性，广泛应用于企业Web应用程序开发和移动应用程序开发。

Java语言和一般编译器以及直译的区别在于，Java首先将源代码转换为字节码，然后将其转换为JVM的可执行文件，JVM可以在各种不同的JVM上运行。因此，实现了它的跨平台特性。虽然这使得Java在早期非常缓慢，但是随着Java的开发，它已经得到了改进。

## 2.6 MVVM模式

MVVM是Model-View-ViewModel的简写。它本质上就是MVC 的改进版。MVVM 就是将其中的View 的状态和行为抽象化，让我们将视图 UI 和业务逻辑分开。当然这些事 ViewModel 已经帮我们做了，它可以取出 Model 的数据同时帮忙处理 View 中由于需要展示内容而涉及的业务逻辑。微软的WPF带来了新的技术体验，如Silverlight、[音频](https://baike.baidu.com/item/%E9%9F%B3%E9%A2%91/1197465" \t "https://baike.baidu.com/item/MVVM/_blank)、[视频](https://baike.baidu.com/item/%E8%A7%86%E9%A2%91/321962" \t "https://baike.baidu.com/item/MVVM/_blank)、[3D](https://baike.baidu.com/item/3D/25017" \t "https://baike.baidu.com/item/MVVM/_blank)、[动画](https://baike.baidu.com/item/%E5%8A%A8%E7%94%BB/206564" \t "https://baike.baidu.com/item/MVVM/_blank)……，这导致了软件UI层更加细节化、可定制化。同时，在技术层面，WPF也带来了 诸如Binding、Dependency Property、Routed Events、Command、DataTemplate、ControlTemplate等新特性。MVVM（Model-View-ViewModel）框架的由来便是MVP（Model-View-Presenter）[模式](https://baike.baidu.com/item/%E6%A8%A1%E5%BC%8F/700029" \t "https://baike.baidu.com/item/MVVM/_blank)与WPF结合的应用方式时发展演变过来的一种新型架构[框架](https://baike.baidu.com/item/%E6%A1%86%E6%9E%B6/1212667" \t "https://baike.baidu.com/item/MVVM/_blank)。它立足于原有MVP框架并且把WPF的新特性糅合进去，以应对客户日益复杂的需求变化。

# **第3章 系统分析**

## 3.1 需求分析

后疫情时代高校宿舍管理系统主要是主要是借助计算机，对轮播图、资源管理、系统用户等进行管理，更好存储所有数据信息及快速方便的检索功能。对系统的各个模块是通过许多今天的发达系统做出合理的分析来确定考虑用户的可操作性，遵循开发的系统优化的原则，经过全面的调查和研究。

系统所要实现的功能分析，对于现在网络方便的管理，系统要实现用户可以直接在平台上进行查看所有数据信息，根据需求可以进行在线添加，删除或修改信息，这样既能节省时间，不用再像传统的方式耽误时间，真的很难去满足用户的各种需求。所以后疫情时代高校宿舍管理系统小程序的开发不仅仅是能满足用户的需求，还能提高管理员的工作效率，减少原有不必要的工作量。

## 3.2 系统可行性分析

### 3.2.1技术可行性

该系统在Windows操作系统中进行开发，并且目前PC机的性能已经可以胜任普通网站的web服务器。系统开发所使用的技术也都是自身所具有的，也是当下广泛应用的技术之一。

系统的开发环境和配置都是可以自行安装的，系统使用JSP开发工具，使用比较成熟的Mysql数据库进行对系统前台及后台的数据交互，根据技术语言对数据库，结合需求进行修改维护，可以使得网站运行更具有稳定性和安全性，从而完成实现网站的开发。

（1）硬件可行性分析

后疫情时代高校宿舍管理系统的设计对于所使用的计算机没有什么硬性的要求，计算机只要可以正常的使用进行代码的编写及页面设计就可行，主要是对于服务器有些要求，对于平台搭建完成要上传的服务器是有一定的要求的，服务器必须选择安全性比较高的，然后就是在打开网站必须顺畅，不能停顿太长时间；性价比高；安全性高。

（2）软件可行性分析

开发整个系统使用的是云计算，流量的可扩展性和基于流量的智能调整云计算的优点就是流量的可扩展性和基于流量的智能调整，保障系统的安全及数据信息的及时备份。

因此，我们从两个方面进行了可行性研究，可以看出系统的开发没有问题。

### 3.2.2经济可行性

在后疫情时代高校宿舍管理系统小程序开发之前所做的市场调研及相关的管理系统，都是没有任何费用的，都是通过开发者自己的努力，所有的工作的都是自己亲力亲为，在碰到自己比较难以解决的问题，大多是通过同学和指导老师的帮助进行相关信息的解决，所以对于后疫情时代高校宿舍管理系统小程序的开发在经济上是完全可行的，没有任何费用支出的。

使用比较成熟的技术，系统是基于B/S模式、java技术，客户端采用Uni-app 、采用Mysql数据库储存数据。所以系统在开发人力、财力要求不高，具有经济可行性。

### 3.2.3操作可行性

可操作性主要是对后疫情时代高校宿舍管理系统小程序设计完成后，用户的使用体验度，以及管理员可以通过系统随时管理相关的数据信息，并且对于管理员、学生用户、宿管用户、辅导员四个用户角色，都可以简单明了的进入到自己的系统界面，通过界面导航菜单可以简单明了地操作功能模块，方便用户信息的操作需求和管理员管理数据信息，对于系统的操作，不需要专业人员都可以直接进行功能模块的操作管理，所以在系统的可操作性是完全可以的。本系统的操作使用的也是界面窗口进行登录，所以操作人员只要会简单的电脑操作就完全可以的。

## 3.3 项目设计目标与原则

1、关于后疫情时代高校宿舍管理系统小程序的基本要求

（1）功能要求：后台首页、管理员（个人资料、修改密码）、轮播图（轮播图管理）、资源管理（校园资讯、资讯分类）、系统用户（管理员、学生用户、宿管用户、辅导员）、模块管理（入住申请、请假申请、报修申请、健康上报、晚归申请、留言反馈、通知公告）等功能模块。

（2）性能：在不同操作系统上均能无差错实现在不同类型的用户登入相应界面后能不出差错、方便地进行预期操作。

（3）安全与保密要求：用户都必须通过身份验证才能进入系统，并且用户的权限也需要根据用户的类型进行限定。

（4）环境要求：支持多种平台，可在Windows系列、Vista系统等多种操作系统下使用。

2、开发目标

后疫情时代高校宿舍管理系统小程序的主要开发目标如下：

（1）实现管理系统信息关系的系统化、规范化和自动化；

（2）减少维护人员的工作量以及实现用户对信息的控制和管理。

（3）方便查询信息及管理信息等；

（4）通过网络操作，改善处理问题的效率，提高操作人员利用率；

（5）考虑到用户多样性特点，要求界面简单，操作简便。

3、设计原则

本系统采用Java技术，Mysql数据库开发，充分保证了系统稳定性、完整性。

后疫情时代高校宿舍管理系统小程序的设计与实现的设计思想如下：

1. 操作简单方便、系统界面安全良、简单明了的页面布局、方便查询相关信息。

2、即时可见：对后疫情时代高校宿舍管理系统小程序信息的处理将可以实时查询到，从而实现“即时发布、即时见效”的系统功能。

3、功能的完善性：拥有后台首页、管理员（个人资料、修改密码）、轮播图（轮播图管理）、资源管理（校园资讯、资讯分类）、系统用户（管理员、学生用户、宿管用户、辅导员）、模块管理（入住申请、请假申请、报修申请、健康上报、晚归申请、留言反馈、通知公告）等模块的修改维护操作。

## 3.4系统流程分析

### 3.4.1操作流程

系统登录流程图，如图所示：



图3-1登录流程图

### 3.4.2添加信息流程

添加信息流程图，如图所示：



图3-2添加信息流程图

### 3.4.3删除信息流程

删除信息流程图，如图所示：



图3-3删除信息流程图

# **第4章 系统设计**

## 4.1 系统体系结构

本系统选择B/S结构(Browser/Server，浏览器/服务器结构)和基于Web服务两种模式。适合在互联网上进行操作，只要用户能连网，任何时间、任何地点都可以进行系统的操作使用。系统工作原理图如图4-1所示：



图4-1 系统工作原理图

## 4.2系统结构设计

整个系统是由多个功能模块组合而成的，要将所有的功能模块都一一列举出来，然后进行逐个的功能设计，使得每一个模块都有相对应的功能设计，然后进行系统整体的设计。

本系统结构图如图4-2所示。



图4-2 系统功能结构图

## 4.3开发流程设计

系统流程的分析是通过调查系统所涉及问题的识别、可行性、可操作性、系统分析处理能力等具体环节来调节、整理系统的设计方案以确保系统能达到理想的状态。这些操作都要从注册、登录处着眼进行一系列的流程测试保证数据库的完整，从而把控系统所涉及信息管理的安全、保证信息输入、输出正常转换。然后，通过实际操作完成流程图的绘制工作。

后疫情时代高校宿舍管理系统小程序的开发对管理模块和系统使用的数据库进行分析，编写代码，系统测试，如图4-4所示。



图4-4开发系统流程图

## 4.4 数据库设计原则

学习编程，我们都知道数据库设计是基于需要设计的系统功能，我们需要建立一个数据库关系模型，用于存储数据信息，这样当我们在程序中时，就没有必要为程序页面添加数据，从而提高系统的效率。数据库存储了很多信息，可以说是信息管理系统的核心和基础，数据库还提供了添加、删除、修改和检查，使系统能够快速找到自己想要的信息，而不是在程序代码中找到。数据库中信息表的每个部分根据一定的关系精确地组合，排列和组合成数据表。

通过后疫情时代高校宿舍管理系统小程序的功能进行规划分成几个实体信息，实体信息将通过ER图进行说明，本系统的主要实体图如下：

管理员信息属性图如图4-5所示。



图4-5 管理员信息实体属性图

系统的E-R图如下图所示：



图4-6系统E-R图

## 4.5 数据表

将数据库概念设计的E-R图转换为关系数据库。在关系数据库中，数据关系由数据表组成，但是表的结构表现在表的字段上。

表access\_token (登陆访问时长)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | token\_id | int | 10 | 0 | N | Y |  | 临时访问牌ID |
| 2 | token | varchar | 64 | 0 | Y | N |  | 临时访问牌 |
| 3 | info | text | 65535 | 0 | Y | N |  |  |
| 4 | maxage | int | 10 | 0 | N | N | 2 | 最大寿命：默认2小时 |
| 5 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 6 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |
| 7 | user\_id | int | 10 | 0 | N | N | 0 | 用户编号: |

表article (文章：用于内容管理系统的文章)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | article\_id | mediumint | 8 | 0 | N | Y |  | 文章id：[0,8388607] |
| 2 | title | varchar | 125 | 0 | N | Y |  | 标题：[0,125]用于文章和html的title标签中 |
| 3 | type | varchar | 64 | 0 | N | N | 0 | 文章分类：[0,1000]用来搜索指定类型的文章 |
| 4 | hits | int | 10 | 0 | N | N | 0 | 点击数：[0,1000000000]访问这篇文章的人次 |
| 5 | praise\_len | int | 10 | 0 | N | N | 0 | 点赞数 |
| 6 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 7 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |
| 8 | source | varchar | 255 | 0 | Y | N |  | 来源：[0,255]文章的出处 |
| 9 | url | varchar | 255 | 0 | Y | N |  | 来源地址：[0,255]用于跳转到发布该文章的网站 |
| 10 | tag | varchar | 255 | 0 | Y | N |  | 标签：[0,255]用于标注文章所属相关内容，多个标签用空格隔开 |
| 11 | content | longtext | 2147483647 | 0 | Y | N |  | 正文：文章的主体内容 |
| 12 | img | varchar | 255 | 0 | Y | N |  | 封面图 |
| 13 | description | text | 65535 | 0 | Y | N |  | 文章描述 |

表article\_type (文章分类)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | type\_id | smallint | 5 | 0 | N | Y |  | 分类ID：[0,10000] |
| 2 | display | smallint | 5 | 0 | N | N | 100 | 显示顺序：[0,1000]决定分类显示的先后顺序 |
| 3 | name | varchar | 16 | 0 | N | N |  | 分类名称：[2,16] |
| 4 | father\_id | smallint | 5 | 0 | N | N | 0 | 上级分类ID：[0,32767] |
| 5 | description | varchar | 255 | 0 | Y | N |  | 描述：[0,255]描述该分类的作用 |
| 6 | icon | text | 65535 | 0 | Y | N |  | 分类图标： |
| 7 | url | varchar | 255 | 0 | Y | N |  | 外链地址：[0,255]如果该分类是跳转到其他网站的情况下，就在该URL上设置 |
| 8 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 9 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |

表auth (用户权限管理)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | auth\_id | int | 10 | 0 | N | Y |  | 授权ID： |
| 2 | user\_group | varchar | 64 | 0 | Y | N |  | 用户组： |
| 3 | mod\_name | varchar | 64 | 0 | Y | N |  | 模块名： |
| 4 | table\_name | varchar | 64 | 0 | Y | N |  | 表名： |
| 5 | page\_title | varchar | 255 | 0 | Y | N |  | 页面标题： |
| 6 | path | varchar | 255 | 0 | Y | N |  | 路由路径： |
| 7 | position | varchar | 32 | 0 | Y | N |  | 位置： |
| 8 | mode | varchar | 32 | 0 | N | N | \_blank | 跳转方式： |
| 9 | add | tinyint | 3 | 0 | N | N | 1 | 是否可增加： |
| 10 | del | tinyint | 3 | 0 | N | N | 1 | 是否可删除： |
| 11 | set | tinyint | 3 | 0 | N | N | 1 | 是否可修改： |
| 12 | get | tinyint | 3 | 0 | N | N | 1 | 是否可查看： |
| 13 | field\_add | text | 65535 | 0 | Y | N |  | 添加字段： |
| 14 | field\_set | text | 65535 | 0 | Y | N |  | 修改字段： |
| 15 | field\_get | text | 65535 | 0 | Y | N |  | 查询字段： |
| 16 | table\_nav\_name | varchar | 500 | 0 | Y | N |  | 跨表导航名称： |
| 17 | table\_nav | varchar | 500 | 0 | Y | N |  | 跨表导航： |
| 18 | option | text | 65535 | 0 | Y | N |  | 配置： |
| 19 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 20 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |

表check\_in\_application (入住申请)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | check\_in\_application\_id | int | 10 | 0 | N | Y |  | 入住申请ID |
| 2 | student\_no | int | 10 | 0 | Y | N | 0 | 学生学号 |
| 3 | student\_name | varchar | 64 | 0 | Y | N |  | 学生姓名 |
| 4 | instructor | int | 10 | 0 | Y | N | 0 | 辅导员 |
| 5 | building\_name | varchar | 64 | 0 | Y | N |  | 楼栋名称 |
| 6 | dormitory\_room\_number | varchar | 64 | 0 | Y | N |  | 宿舍房号 |
| 7 | date\_of\_check\_in | date | 10 | 0 | Y | N |  | 入住日期 |
| 8 | check\_in\_instructions | text | 65535 | 0 | Y | N |  | 入住说明 |
| 9 | examine\_state | varchar | 16 | 0 | N | N | 未审核 | 审核状态 |
| 10 | examine\_reply | varchar | 16 | 0 | Y | N |  | 审核回复 |
| 11 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 12 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 13 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表collect (收藏)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | collect\_id | int | 10 | 0 | N | Y |  | 收藏ID： |
| 2 | user\_id | int | 10 | 0 | N | N | 0 | 收藏人ID： |
| 3 | source\_table | varchar | 255 | 0 | Y | N |  | 来源表： |
| 4 | source\_field | varchar | 255 | 0 | Y | N |  | 来源字段： |
| 5 | source\_id | int | 10 | 0 | N | N | 0 | 来源ID： |
| 6 | title | varchar | 255 | 0 | Y | N |  | 标题： |
| 7 | img | varchar | 255 | 0 | Y | N |  | 封面： |
| 8 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 9 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |

表comment (评论)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | comment\_id | int | 10 | 0 | N | Y |  | 评论ID： |
| 2 | user\_id | int | 10 | 0 | N | N | 0 | 评论人ID： |
| 3 | reply\_to\_id | int | 10 | 0 | N | N | 0 | 回复评论ID：空为0 |
| 4 | content | longtext | 2147483647 | 0 | Y | N |  | 内容： |
| 5 | nickname | varchar | 255 | 0 | Y | N |  | 昵称： |
| 6 | avatar | varchar | 255 | 0 | Y | N |  | 头像地址：[0,255] |
| 7 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 8 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |
| 9 | source\_table | varchar | 255 | 0 | Y | N |  | 来源表： |
| 10 | source\_field | varchar | 255 | 0 | Y | N |  | 来源字段： |
| 11 | source\_id | int | 10 | 0 | N | N | 0 | 来源ID： |

表health\_reporting (健康上报)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | health\_reporting\_id | int | 10 | 0 | N | Y |  | 健康上报ID |
| 2 | student\_no | int | 10 | 0 | Y | N | 0 | 学生学号 |
| 3 | student\_name | varchar | 64 | 0 | Y | N |  | 学生姓名 |
| 4 | instructor | int | 10 | 0 | Y | N | 0 | 辅导员 |
| 5 | reporting\_date | date | 10 | 0 | Y | N |  | 上报日期 |
| 6 | temperature\_today | varchar | 64 | 0 | Y | N |  | 今日体温 |
| 7 | health\_code | varchar | 255 | 0 | Y | N |  | 健康码 |
| 8 | is\_it\_uncomfortable | varchar | 64 | 0 | Y | N |  | 是否不适 |
| 9 | unwell\_symptoms | text | 65535 | 0 | Y | N |  | 不适症状 |
| 10 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 11 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 12 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表hits (用户点击)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | hits\_id | int | 10 | 0 | N | Y |  | 点赞ID： |
| 2 | user\_id | int | 10 | 0 | N | N | 0 | 点赞人： |
| 3 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 4 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |
| 5 | source\_table | varchar | 255 | 0 | Y | N |  | 来源表： |
| 6 | source\_field | varchar | 255 | 0 | Y | N |  | 来源字段： |
| 7 | source\_id | int | 10 | 0 | N | N | 0 | 来源ID： |

表housing\_management\_users (宿管用户)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | housing\_management\_users\_id | int | 10 | 0 | N | Y |  | 宿管用户ID |
| 2 | hanguan\_job\_number | varchar | 64 | 0 | N | N |  | 宿管工号 |
| 3 | name\_of\_the\_housekeeper | varchar | 64 | 0 | Y | N |  | 宿管姓名 |
| 4 | examine\_state | varchar | 16 | 0 | N | N | 已通过 | 审核状态 |
| 5 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 6 | user\_id | int | 10 | 0 | N | N | 0 | 用户ID |
| 7 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 8 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表instructor (辅导员)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | instructor\_id | int | 10 | 0 | N | Y |  | 辅导员ID |
| 2 | trainee\_no | varchar | 64 | 0 | N | N |  | 辅导员工号 |
| 3 | name\_of\_instructor | varchar | 64 | 0 | Y | N |  | 辅导员姓名 |
| 4 | examine\_state | varchar | 16 | 0 | N | N | 已通过 | 审核状态 |
| 5 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 6 | user\_id | int | 10 | 0 | N | N | 0 | 用户ID |
| 7 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 8 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表late\_return\_application (晚归申请)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | late\_return\_application\_id | int | 10 | 0 | N | Y |  | 晚归申请ID |
| 2 | student\_no | int | 10 | 0 | Y | N | 0 | 学生学号 |
| 3 | student\_name | varchar | 64 | 0 | Y | N |  | 学生姓名 |
| 4 | housemaster | int | 10 | 0 | Y | N | 0 | 宿管 |
| 5 | dormitory\_room\_number | varchar | 64 | 0 | Y | N |  | 宿舍房号 |
| 6 | late\_return\_time | datetime | 19 | 0 | Y | N |  | 晚归时间 |
| 7 | reasons\_for\_late\_return | text | 65535 | 0 | Y | N |  | 晚归原因 |
| 8 | examine\_state | varchar | 16 | 0 | N | N | 未审核 | 审核状态 |
| 9 | examine\_reply | varchar | 16 | 0 | Y | N |  | 审核回复 |
| 10 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 11 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 12 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表leave\_application (请假申请)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | leave\_application\_id | int | 10 | 0 | N | Y |  | 请假申请ID |
| 2 | student\_no | int | 10 | 0 | Y | N | 0 | 学生学号 |
| 3 | student\_name | varchar | 64 | 0 | Y | N |  | 学生姓名 |
| 4 | instructor | int | 10 | 0 | Y | N | 0 | 辅导员 |
| 5 | leave\_type | varchar | 64 | 0 | Y | N |  | 请假类型 |
| 6 | start\_time | datetime | 19 | 0 | Y | N |  | 开始时间 |
| 7 | leave\_days | varchar | 64 | 0 | Y | N |  | 请假天数 |
| 8 | end\_time | datetime | 19 | 0 | Y | N |  | 结束时间 |
| 9 | reason\_for\_taking\_leave | text | 65535 | 0 | Y | N |  | 请假原因 |
| 10 | examine\_state | varchar | 16 | 0 | N | N | 未审核 | 审核状态 |
| 11 | examine\_reply | varchar | 16 | 0 | Y | N |  | 审核回复 |
| 12 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 13 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 14 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表message\_feedback (留言反馈)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | message\_feedback\_id | int | 10 | 0 | N | Y |  | 留言反馈ID |
| 2 | student\_no | int | 10 | 0 | Y | N | 0 | 学生学号 |
| 3 | student\_name | varchar | 64 | 0 | Y | N |  | 学生姓名 |
| 4 | student\_phone | varchar | 64 | 0 | Y | N |  | 学生电话 |
| 5 | title | varchar | 64 | 0 | Y | N |  | 标题 |
| 6 | message\_content | text | 65535 | 0 | Y | N |  | 留言内容 |
| 7 | feedback | text | 65535 | 0 | Y | N |  | 反馈意见 |
| 8 | message\_reply | text | 65535 | 0 | Y | N |  | 留言回复 |
| 9 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 10 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 11 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表notice\_announcement (通知公告)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | notice\_announcement\_id | int | 10 | 0 | N | Y |  | 通知公告ID |
| 2 | publisher | int | 10 | 0 | Y | N | 0 | 发布人 |
| 3 | title | varchar | 64 | 0 | Y | N |  | 标题 |
| 4 | release\_time | date | 10 | 0 | Y | N |  | 发布时间 |
| 5 | publishing\_content | text | 65535 | 0 | Y | N |  | 内容 |
| 6 | related\_images | varchar | 255 | 0 | Y | N |  | 相关图片 |
| 7 | hits | int | 10 | 0 | N | N | 0 | 点击数 |
| 8 | praise\_len | int | 10 | 0 | N | N | 0 | 点赞数 |
| 9 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 10 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 11 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表praise (点赞)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | praise\_id | int | 10 | 0 | N | Y |  | 点赞ID： |
| 2 | user\_id | int | 10 | 0 | N | N | 0 | 点赞人： |
| 3 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 4 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |
| 5 | source\_table | varchar | 255 | 0 | Y | N |  | 来源表： |
| 6 | source\_field | varchar | 255 | 0 | Y | N |  | 来源字段： |
| 7 | source\_id | int | 10 | 0 | N | N | 0 | 来源ID： |
| 8 | status | bit | 1 | 0 | N | N | 1 | 点赞状态:1为点赞，0已取消 |

表repair\_application (报修申请)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | repair\_application\_id | int | 10 | 0 | N | Y |  | 报修申请ID |
| 2 | student\_no | int | 10 | 0 | Y | N | 0 | 学生学号 |
| 3 | student\_name | varchar | 64 | 0 | Y | N |  | 学生姓名 |
| 4 | housemaster | int | 10 | 0 | Y | N | 0 | 宿管 |
| 5 | building\_name | varchar | 64 | 0 | Y | N |  | 楼栋名称 |
| 6 | dormitory\_room\_number | varchar | 64 | 0 | Y | N |  | 宿舍房号 |
| 7 | items\_for\_repair | varchar | 64 | 0 | Y | N |  | 报修物品 |
| 8 | repair\_reason | text | 65535 | 0 | Y | N |  | 报修原因 |
| 9 | examine\_state | varchar | 16 | 0 | N | N | 未审核 | 审核状态 |
| 10 | examine\_reply | varchar | 16 | 0 | Y | N |  | 审核回复 |
| 11 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 12 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 13 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表slides (轮播图)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | slides\_id | int | 10 | 0 | N | Y |  | 轮播图ID： |
| 2 | title | varchar | 64 | 0 | Y | N |  | 标题： |
| 3 | content | varchar | 255 | 0 | Y | N |  | 内容： |
| 4 | url | varchar | 255 | 0 | Y | N |  | 链接： |
| 5 | img | varchar | 255 | 0 | Y | N |  | 轮播图： |
| 6 | hits | int | 10 | 0 | N | N | 0 | 点击量： |
| 7 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 8 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |

表student\_users (学生用户)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | student\_users\_id | int | 10 | 0 | N | Y |  | 学生用户ID |
| 2 | student\_no | varchar | 64 | 0 | N | N |  | 学生学号 |
| 3 | student\_name | varchar | 64 | 0 | Y | N |  | 学生姓名 |
| 4 | examine\_state | varchar | 16 | 0 | N | N | 已通过 | 审核状态 |
| 5 | recommend | int | 10 | 0 | N | N | 0 | 智能推荐 |
| 6 | user\_id | int | 10 | 0 | N | N | 0 | 用户ID |
| 7 | create\_time | datetime | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间 |
| 8 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间 |

表upload (文件上传)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | upload\_id | int | 10 | 0 | N | Y |  | 上传ID |
| 2 | name | varchar | 64 | 0 | Y | N |  | 文件名 |
| 3 | path | varchar | 255 | 0 | Y | N |  | 访问路径 |
| 4 | file | varchar | 255 | 0 | Y | N |  | 文件路径 |
| 5 | display | varchar | 255 | 0 | Y | N |  | 显示顺序 |
| 6 | father\_id | int | 10 | 0 | Y | N | 0 | 父级ID |
| 7 | dir | varchar | 255 | 0 | Y | N |  | 文件夹 |
| 8 | type | varchar | 32 | 0 | Y | N |  | 文件类型 |

表user (用户账户：用于保存用户登录信息)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | user\_id | mediumint | 8 | 0 | N | Y |  | 用户ID：[0,8388607]用户获取其他与用户相关的数据 |
| 2 | state | smallint | 5 | 0 | N | N | 1 | 账户状态：[0,10](1可用|2异常|3已冻结|4已注销) |
| 3 | user\_group | varchar | 32 | 0 | Y | N |  | 所在用户组：[0,32767]决定用户身份和权限 |
| 4 | login\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 上次登录时间： |
| 5 | phone | varchar | 11 | 0 | Y | N |  | 手机号码：[0,11]用户的手机号码，用于找回密码时或登录时 |
| 6 | phone\_state | smallint | 5 | 0 | N | N | 0 | 手机认证：[0,1](0未认证|1审核中|2已认证) |
| 7 | username | varchar | 16 | 0 | N | N |  | 用户名：[0,16]用户登录时所用的账户名称 |
| 8 | nickname | varchar | 16 | 0 | Y | N |  | 昵称：[0,16] |
| 9 | password | varchar | 64 | 0 | N | N |  | 密码：[0,32]用户登录所需的密码，由6-16位数字或英文组成 |
| 10 | email | varchar | 64 | 0 | Y | N |  | 邮箱：[0,64]用户的邮箱，用于找回密码时或登录时 |
| 11 | email\_state | smallint | 5 | 0 | N | N | 0 | 邮箱认证：[0,1](0未认证|1审核中|2已认证) |
| 12 | avatar | varchar | 255 | 0 | Y | N |  | 头像地址：[0,255] |
| 13 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |

表user\_group (用户组：用于用户前端身份和鉴权)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 名称 | 数据类型 | 长度 | 小数位 | 允许空值 | 主键 | 默认值 | 说明 |
| 1 | group\_id | mediumint | 8 | 0 | N | Y |  | 用户组ID：[0,8388607] |
| 2 | display | smallint | 5 | 0 | N | N | 100 | 显示顺序：[0,1000] |
| 3 | name | varchar | 16 | 0 | N | N |  | 名称：[0,16] |
| 4 | description | varchar | 255 | 0 | Y | N |  | 描述：[0,255]描述该用户组的特点或权限范围 |
| 5 | source\_table | varchar | 255 | 0 | Y | N |  | 来源表： |
| 6 | source\_field | varchar | 255 | 0 | Y | N |  | 来源字段： |
| 7 | source\_id | int | 10 | 0 | N | N | 0 | 来源ID： |
| 8 | register | smallint | 5 | 0 | Y | N | 0 | 注册位置: |
| 9 | create\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 创建时间： |
| 10 | update\_time | timestamp | 19 | 0 | N | N | CURRENT\_TIMESTAMP | 更新时间： |

# 第5章 系统详细设计

## 5.1管理员功能模块

管理员登录后疫情时代高校宿舍管理系统，在登录页面输入用户名、密码、图片验证等进行登录，如图所示。



图5-1管理员登录界面图

管理员进入系统后台可以查看：后台首页、管理员（个人资料、修改密码）、轮播图（轮播图管理）、资源管理（校园资讯、资讯分类）、系统用户（管理员、学生用户、宿管用户、辅导员）、模块管理（入住申请、请假申请、报修申请、健康上报、晚归申请、留言反馈、通知公告）。

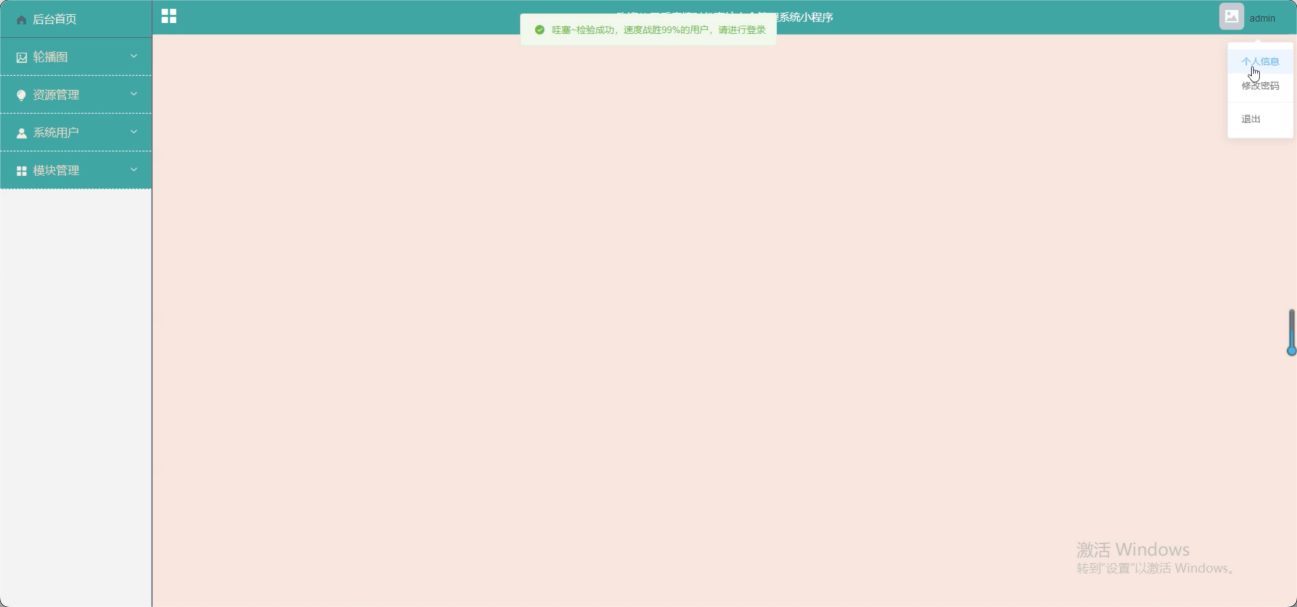


图5-2管理员功能界面图

轮播图，在此页面可以查看轮播图、标题等内容，还可在该页面进行重置、查询、添加、删除等操作。如图所示。

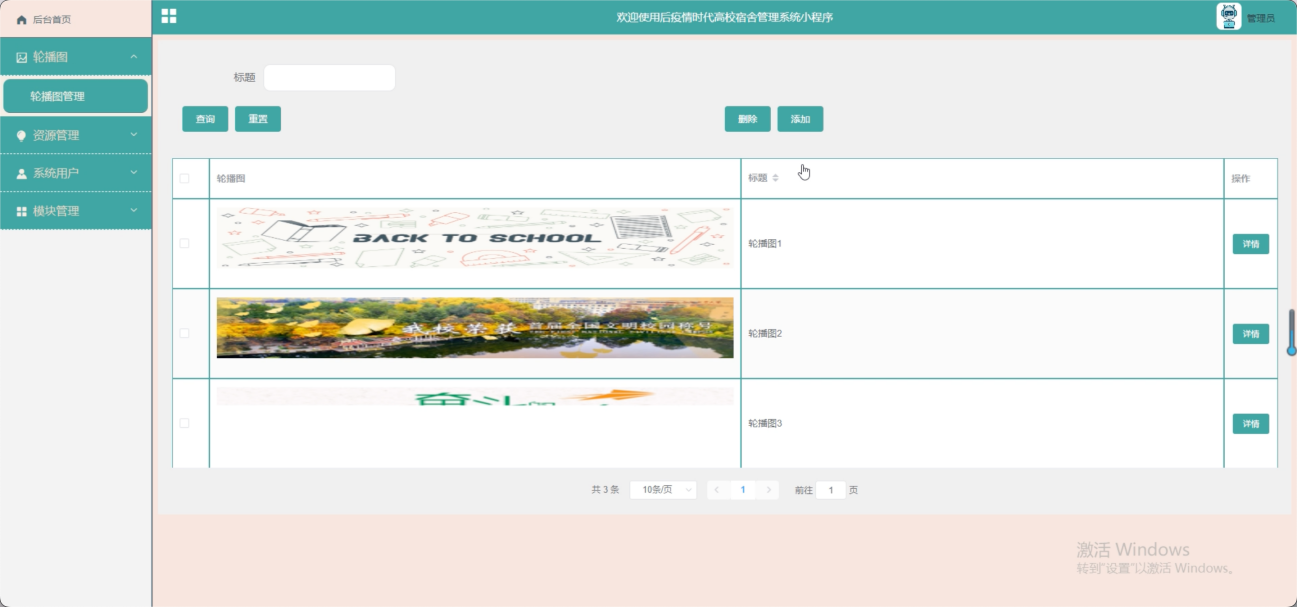


图5-3轮播图管理界面图

资源管理：在此页面有校园资讯和资讯分类两个子模块，可以查看标题、封面图、文章分类、标签等信息，并进行查询、重置、删除、添加等操作，如图所示。

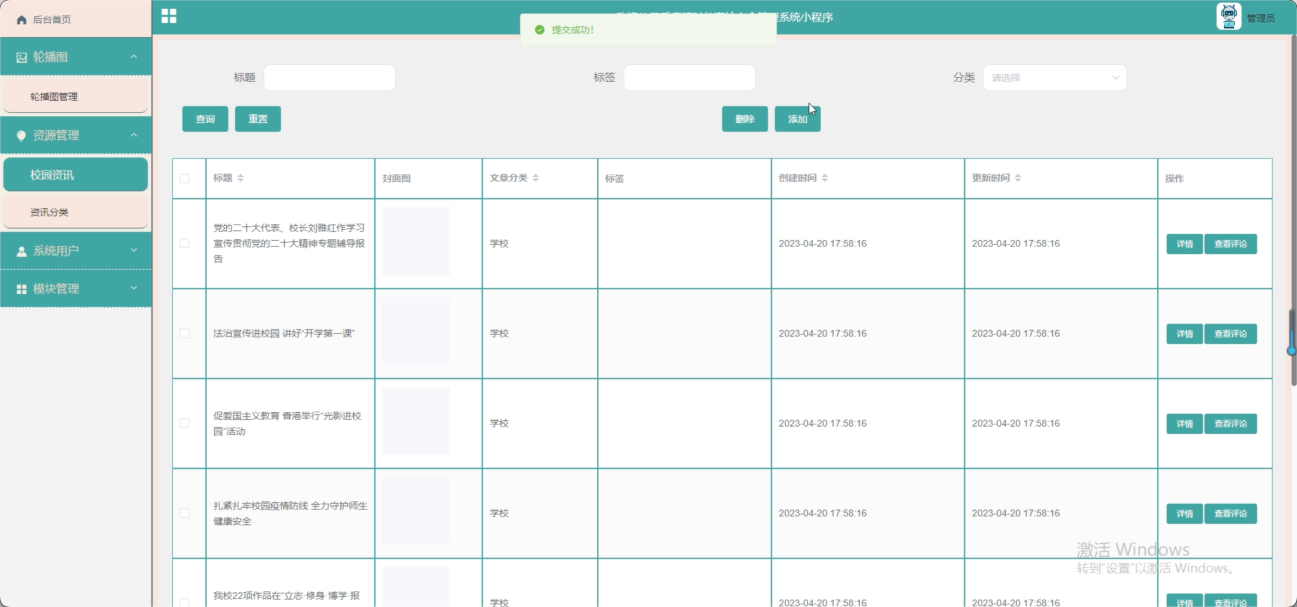


图5-4资讯管理界面图

系统用户：在此页面有管理员、学生用户、宿管用户、辅导员四个子模块，可以查看用户名、创建时间、更新时间等信息并进行查询、重置、删除、添加等操作，如图所示。

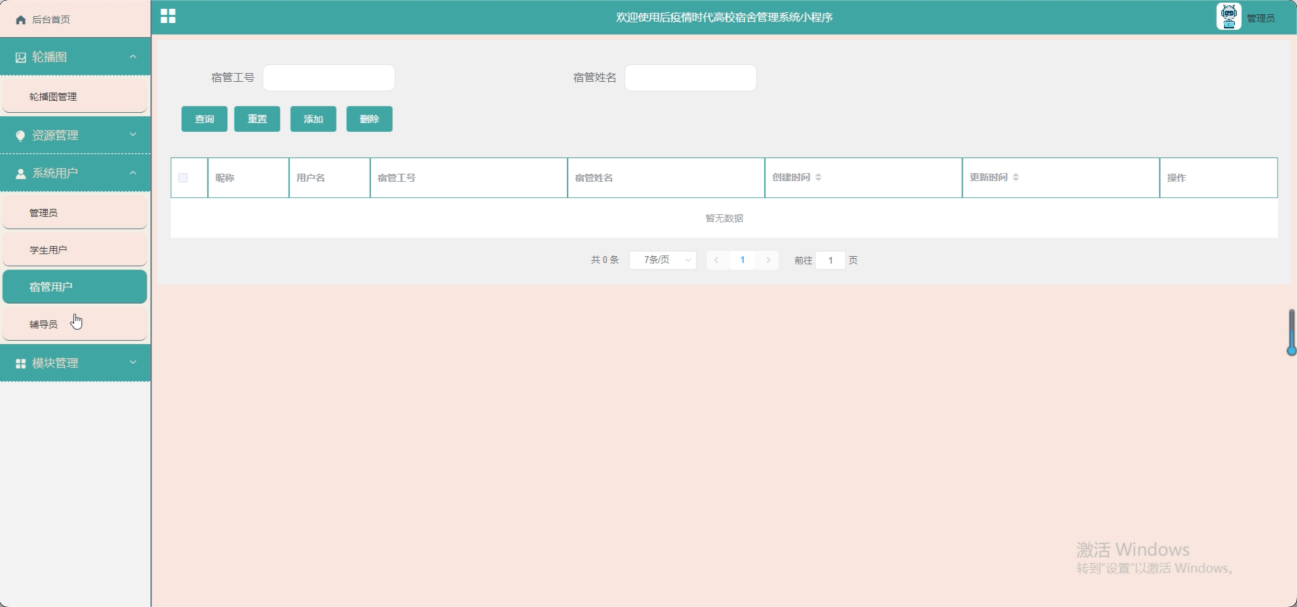


图5-5系统用户界面图

模块管理：在此页面有入住申请、请假申请、报修申请、健康上报、晚归申请、留言反馈、通知公告七个子模块并进行查询、重置、删除、添加等操作，如图所示。

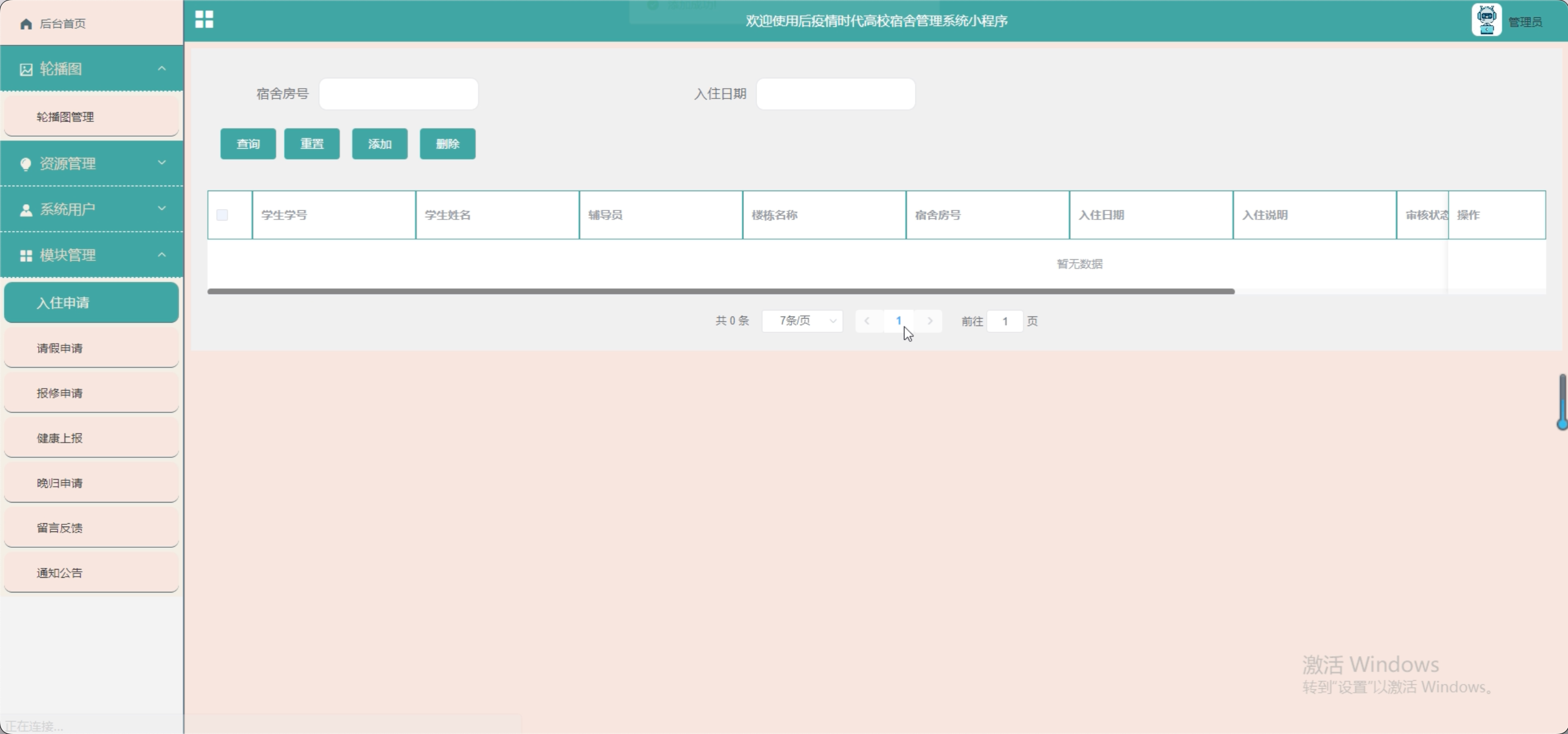


图5-6模板管理界面图

入住申请：可以查看学生姓名、宿舍房号、入住时间等信息并进行查询、重置、删除、添加等操作，如图所示。

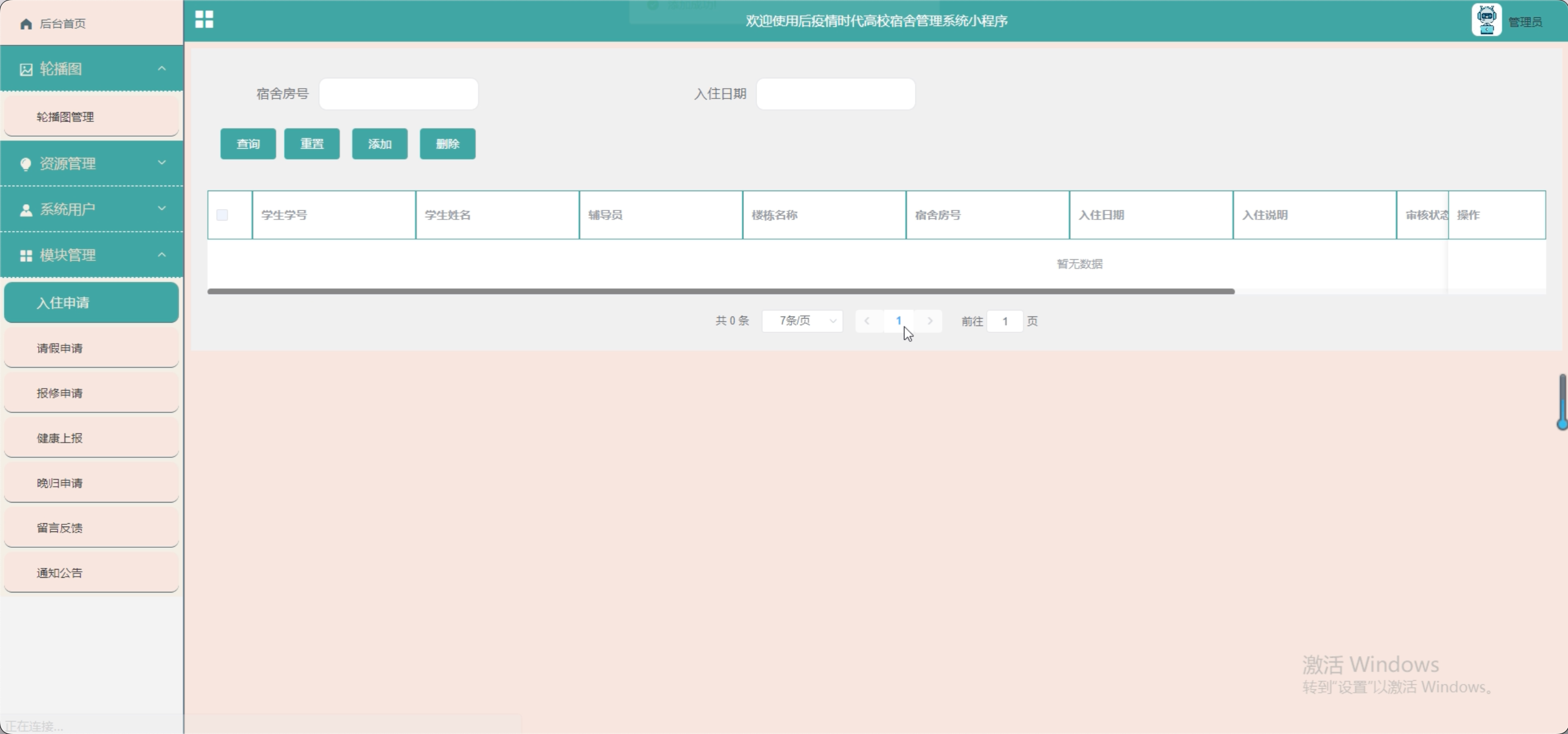


图5-7入住申请界面图

请假申请：可以查看学生姓名、请假类型、开始时间、请假原因等信息并进行查询、重置、删除、添加等操作，如图所示。

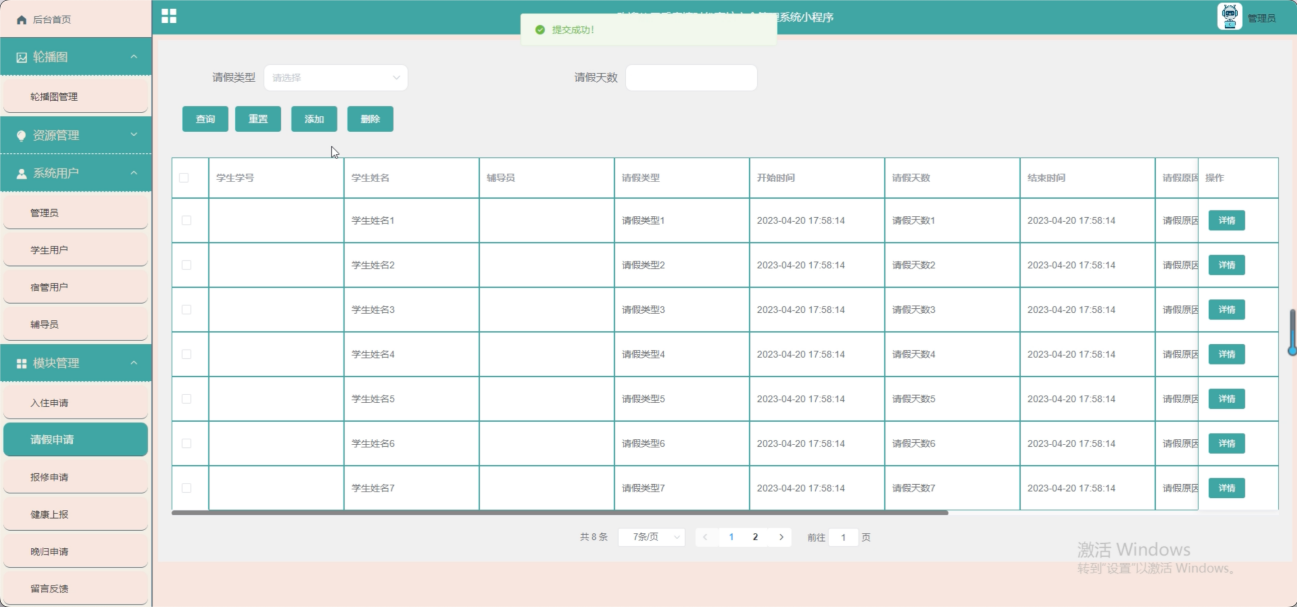


图5-8请假申请界面图

报修申请：可以查看学生姓名、报修物品、审核状态等信息并进行查询、重置、删除、添加等操作，如图所示。

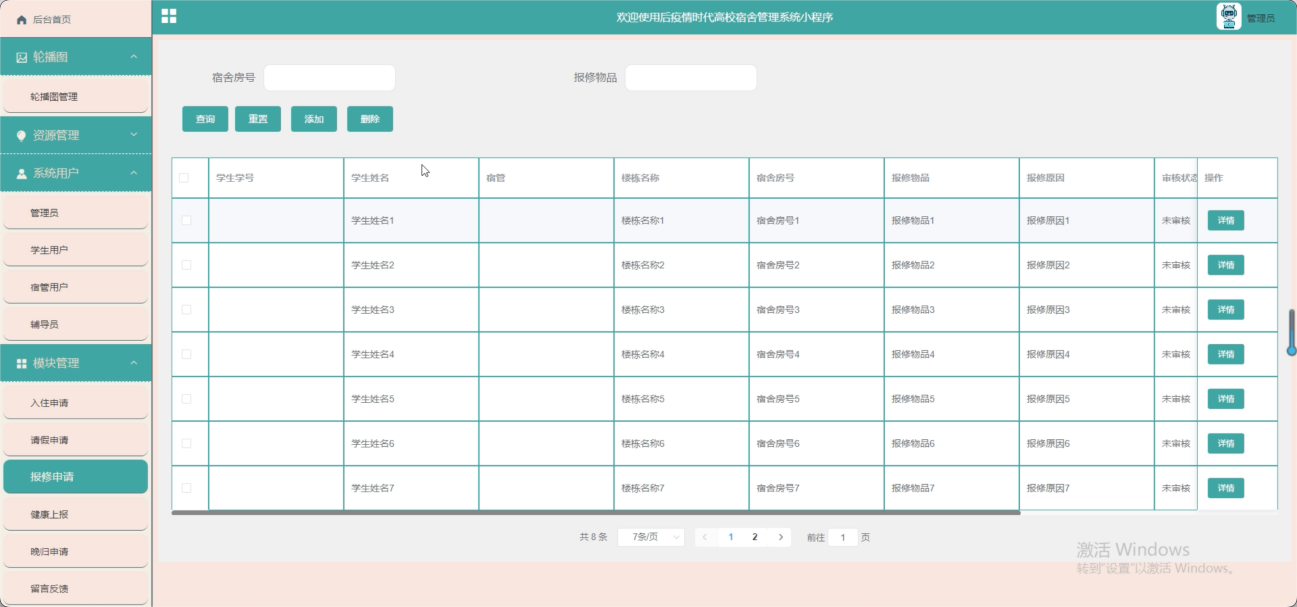


图5-9报修申请界面图

健康上报：可以查看学生姓名、上报日期、是否不适等信息并进行查询、重置、删除、添加等操作，如图所示。

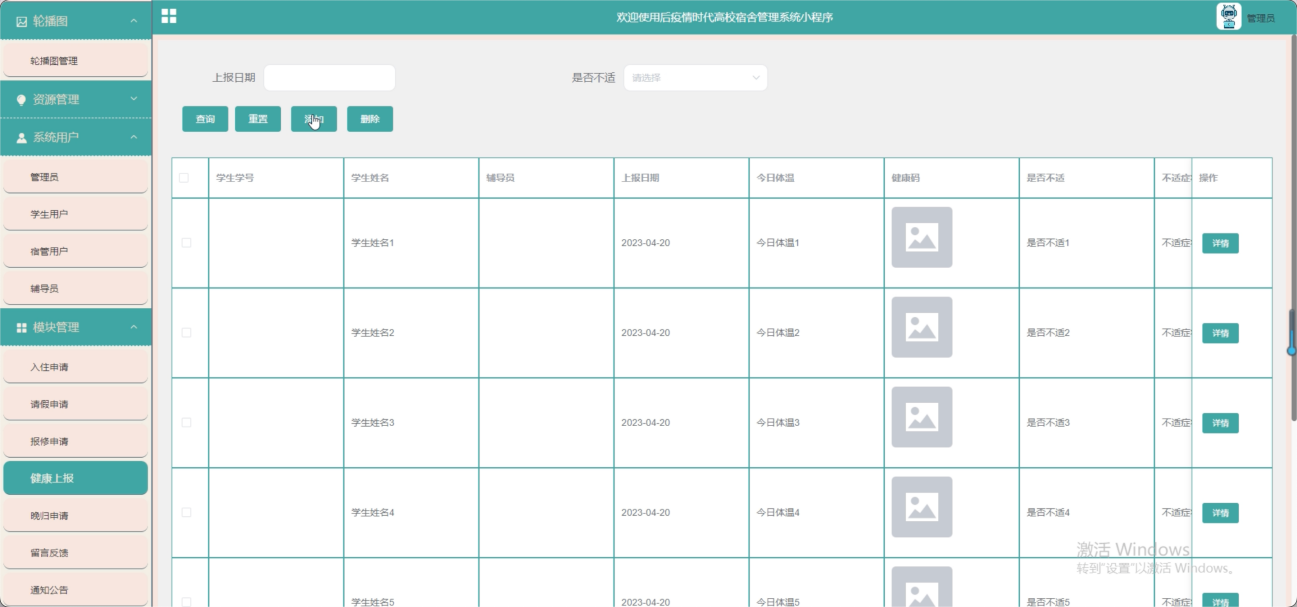


图5-10健康上报界面图

晚归申请：可以查看学生姓名、晚归时间、晚归原因等信息并进行查询、重置、删除、添加等操作，如图所示。

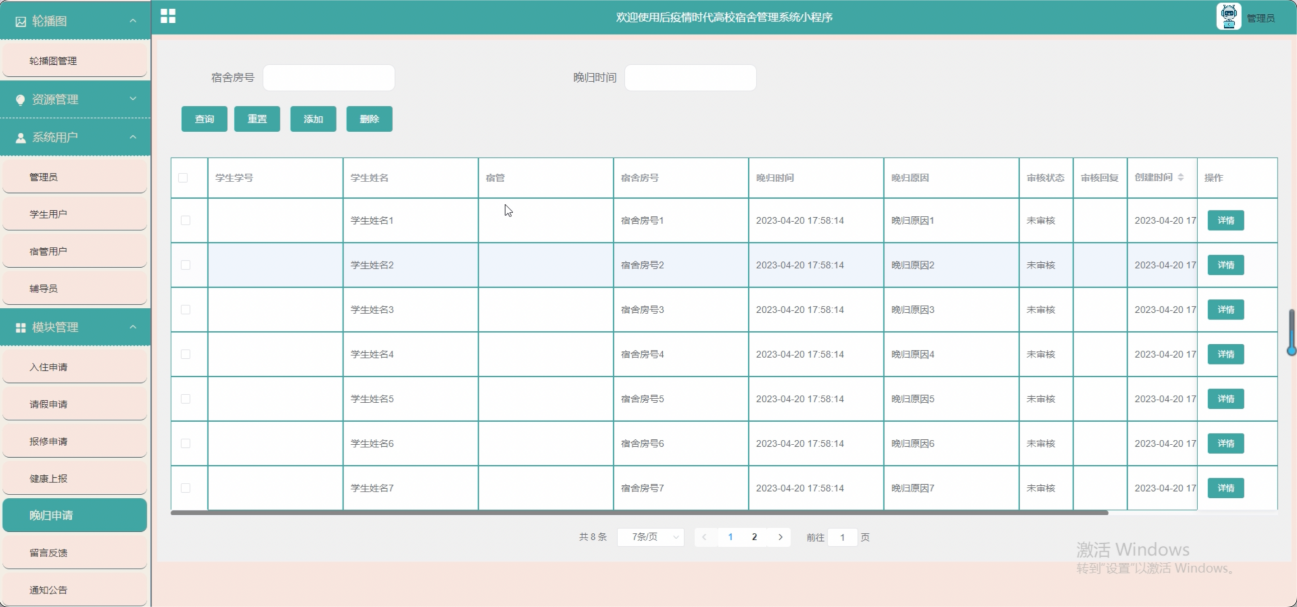


图5-11晚归申请界面图

## 5.2学生用户功能模块

学生用户登录后疫情时代高校宿舍管理系统小程序，在登录页面输入用户名、密码进行登录，如图所示。



图5-12学生用户登录界面图

进入系统可以查看：首页、资讯、我的（基本信息、入住申请、请假申请、报修申请、健康上报、晚归申请、留言反馈、通知公告），还可进行相应的操作，如图所示。



图5-13学生用户功能界面图

入住申请，在该页面可以编辑学生姓名、宿舍房号、入住时间等信息，如图所示。



图5-14入住申请界面图

请假申请，在该页面可以编辑学生姓名、请假类型、开始时间、请假原因等信息，如图所示。



图5-15请假申请界面图

报修申请：在该页面可以编辑学生姓名、报修物品、审核状态等信息，如图所示。



图5-16报修申请界面图

健康上报：在该页面可以完成健康上报，如图所示。



图5-17健康上报界面图

晚归申请：在该页面可以编辑学生姓名、晚归时间、晚归原因等信息，如图所示。



图5-18晚归申请界面图

## 5.3辅导员功能模块

辅导员进入系统后台可以查看：后台首页、首页、资讯、我的（基本信息、学生用户、宿管用户、入住申请、请假申请、健康上报、留言反馈、通知公告），如图所示。



图5-19我的界面图

宿管用户，在此页面可以添加宿管用户的相关信息。如图所示。



图5-20宿管用户添加界面图

健康上报，在此页面可以查看学生的健康上报信息。如图所示。



图5-21健康上报审核界面图

入住申请，在此页面可以查看学生姓名、入住日期等信息，并进行审核等操作。如图所示。



图5-22入住申请审核界面图

通知公告，在此页面可以编辑发布人、标题、内容等信息进行提交，如图所示。



图5-23通知公告界面图

## 5.4宿管功能模块

宿管进入系统后台可以查看：页、资讯、我的（基本信息、报修申请、晚归申请）如图所示。



图5-24宿管功能界面图

报修申请，在该页面可以查看学生姓名、报修物品、审核状态等信息并进行审核等操作，如图所示。



图5-25报修申请审核界面图

晚归申请，在该页面可以查看学生姓名、晚归时间、晚归原因等信息并进行审核等操作，如图所示。



图5-26晚归申请审核界面图

# **第6章 系统测试**

## 6.1系统测试的目的

程序设计不能保证没有错误，这是一个开发过程，在错误或错误的过程中都是难以避免的。虽然这是不可避免的，但我们不能使这些错误始终存在于系统中，错误可能会造成无法估量的后果，如系统崩溃，安全信息泄露，系统无法正常启动等，为了避免这些问题，我们需要测试程序，再测试过程中发现问题，并纠正它们，从而使系统更长时间稳定成熟。本章的作用是发现这些问题，并对其进行修改，虽然耗时费力，但对于长期使用而言是非常重要和必要系统的开发。

软件在设计后必须进行测试，调试过程中使用的方法是软件测试方法。在开发新软件时，系统测试是检查软件是否合格的关键步骤，以及是否符合设计目标的参考。测试主要是查看软件中数据的准确性，正确的操作与否，以及操作的结果，还有哪些方面需要改进。

后疫情时代高校宿舍管理系统小的实现，对于系统中功能模块的实现及操作都必须通过测试进行来评判系统是否可以准确的实现。在后疫情时代高校宿舍管理系统小程序的正式上传使用之前必须做的一步就是系统测试，对于测试发现的错误及时修改处理，保证系统准确无误的供给用户使用。

## 6.2系统测试方法

在对后疫情时代高校宿舍管理系统小程序进行测试的时候在找到问题的情况下必须在第一时间找到解决问题的办法，不要存在侥幸的心理，这样才能让后疫情时代高校宿舍管理系统小程序开发的质量可以过关，并且开发的周期会大大缩短，还有就是在测试时，不要出现重复性的错误，遇到一个错误问题，要将整个后疫情时代高校宿舍管理系统小程序开发所牵扯的该问题都必须一一解决，提高后疫情时代高校宿舍管理系统小程序平台的安全性、稳定性。

白盒测试与黑盒测试是测试中比较常用的两种方法。

①结构测试俗称白盒测试：这种测试是在对程序的处理过程与结构都有详尽谅解的前提下，顺从程序内部的逻辑而完成的系统测试，以确定系统中所有的通路都能够遵照设计要求正常工作，不出现任何偏差。

②功能测试又成黑盒测试：主要是针对程序功能能够按照设计正常实现的一种检测，在程序接口处进行，检测程序手法数据是否正常，与外部信息的交换是否完整。

## 6.3功能测试

用户登录测试：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块名称 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 登录模块 | 用户名：admin 密码：123 | 弹出错误提示，提示密码错误 | 弹出错误提示，提示密码错误 | 通过 |
| 登录模块 | 用户名：123  密码：admin | 弹出错误提示，提示用户名错误 | 弹出错误提示，提示用户名错误 | 通过 |
| 登录模块 | 用户名：admin  密码：admin | 管理员登录成功 | 管理员登录成功 | 通过 |

修改密码测试：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块名称 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 修改密码模块 | 原密码：666  新密码：123  确认密码：123 | 弹出错误提示，提示原密码错误 | 弹出错误提示，提示原密码错误 | 通过 |
| 修改密码模块 | 原密码：admin 新密码：123  确认密码：333 | 弹出错误提示，提示确认密码不一致 | 弹出错误提示，提示确认密码不一致 | 通过 |
| 修改密码模块 | 原密码：admin 新密码：123  确认密码：123 | 密码修改成功 | 密码修改成功 | 通过 |

通过对功能的测试，后疫情时代高校宿舍管理系统小程序的基本功能都是可行的，不管是系统里面的功能，还是界面的设计都是可值得推广宣传的。

结 论

此时项目已经完成，即使实施的时间不是很长，但是这个过程中需要准备很长的一段时间去对系统设计开发所相关技术进行学习。在学习的过程中，我逐渐认识得到了我自身存在的一些不足。对于一些控制是必要的应用技能，能够理解，整个过程中仅仅是一个掌握了常用的性能和控制方法，我觉得挺容易的。从该系统中，系统的分析和设计的调查数据，并且已经经历了几个月，并努力几个月，该系统已经完成。很显然，该系统仍有很多不成熟的地方，在系统设计过程中有许多技术缺陷存在。在设计的过程中也涉及到了很多自己无法解决的问题，主要通过咨询老师解决这些问题，对于毕业设计的圆满完成，需要感谢老师们的指导。系统的开发环境和配置都是可以自行安装的，系统使用Java开发工具，使用比较成熟的Mysql数据库进行对系统前台及后台的数据交互，根据技术语言对数据库，结合需求进行修改维护，可以使得系统运行更具有稳定性和安全性，从而完成实现系统的开发。

回顾毕业设计的整个过程，既付出汗水也收获了很多。虽然经历了各种各样的困难，自己的不断研究探索，系统的实现仍有不足之处。

在以后的学习及工作中，我仍然继续学习计算机方面的技术，让我在后期的平台开发中可以更好更快的实现需求功能。我相信我可以让更多的好工作，做出更大的贡献。

致 谢

后疫情时代高校宿舍管理系统小程序的完成，如何实现的更好，其中付出的努力是很大的，这段时光将会终身难忘。

首先要感谢我的指导老师，谢谢您在设计和论文中给我的指导。在您的细心指导下我才能快速的掌握系统的相关功能，在您的大力帮助下我才能将课本上的知识与自己的项目结合，真正的做到学以致用。感谢您经常牺牲自己的休息时间，利用其丰富的教学和项目经验对我进行指导。

感谢所有教过我的老师，为我倾注了大量的心血，正是你们的谆谆教诲、严谨教学才使我能顺利的完成学业，再此向你们表示深深的感谢。

感谢我的同学们，对我的大力支持及帮助，正是你们不断的帮助、鼓励，给我带来了极大的动力，最终系统可以顺利的运行。我们在交流、谈论的这段时间，将是我未来的财富，我要深深地感谢你们!

毕业在即，在今后的工作和生活中，我会铭记师长们的教诲、同学们的帮助，继续不懈努力和追求，来报答所有支持和帮助过我的人!

# 参考文献

[1]唐瑞明,谭倩芳. 高校宿舍信息化管理系统设计[J]. 电子技术与软件工程,2023,(04):258-262.

[2]Yu Yang. Design and Implementation of Student Information Management System Based on Springboot[J]. Advances in Computer, Signals and Systems,2022,6(6).

[3]Møller Pål. The Prospective Lynch Syndrome Database: background, design, main results and complete MySQL code[J]. Hereditary Cancer in Clinical Practice,2022,20(1).

[4]李洋,胡茵茵,宣宏坤,黄春健,黄泽涛,王宗彤. 智慧宿舍管理信息系统的设计与分析[J]. 无线互联科技,2021,18(20):42-44+49.

[5]Paul Krill. Oracle accelerates MySQL HeatWave queries with machine learning[J]. InfoWorld.com,2021.

[6]Zhao Liang,Gu Hengzhi,Xu Jingxian,Cui Yang,Shuai Chunjiang. Research on Simufact simulation data processing system based on QT and MySQL[J]. Applied Mathematics and Nonlinear Sciences,2021,6(2).

[7]Fuyuan Cheng. Talent Recruitment Management System for Small and Micro Enterprises Based on Springboot Framework[J]. Advances in Educational Technology and Psychology,2021,5(2).

[8]Fatima M. Isiaka,Salihu Abdullahi Audu,Mustafa Ahmed Umar. Developing a fail-safe culture in a cyber environment using MySQL replication technique[J]. International Journal of Crowd Science,2020,4(2).

[9]唐瑞明,李论,陈珊. 高校宿舍管理系统综述[J]. 电子技术与软件工程,2020,(04):64-66.

[10]王青. 高校宿舍智能管理系统设计[J]. 微型电脑应用,2019,35(06):135-137.

[11]陈禹吣,袁鸿燕,张荻汶,刘飞鸿. 基于微信小程序的高校学生宿舍报修管理系统的设计[J]. 电脑知识与技术,2019,15(14):36-37.

[12]李昂. 基于Android和SpringBoot的高校宿舍管理系统设计[J]. 铜陵职业技术学院学报,2019,18(01):76-79.

[13]井月. 高校学生宿舍管理系统功能需求分析——基于学生管理的视角[J]. 中国管理信息化,2018,21(05):140-141.

[14]于佳佳,刘彩云. 基于“互联网+”下高校学生宿舍管理系统的研究[J]. 现代职业教育,2017,(03):116.

[15]万德生. 高校宿舍管理系统的设计与实现[D].吉林大学,2016.

[16]黄伟斌. 高校宿舍管理系统需求分析[J]. 中国管理信息化,2015,18(08):85-86.

[17]齐慧敏. 高校宿舍管理系统的设计与实现[D].西安电子科技大学,2013.

[18]来涛. 高校学生宿舍信息管理系统[J]. 山西师大学报(社会科学版),2013,40(S1):140-141.

[19]辜焕彬. 高校学生宿舍管理系统的架构分析[J]. 黑龙江科技信息,2011,(06):175.

[20]欧仁侠,陈洪斌,祝颖. 高校宿舍管理系统中电子设备的应用研究[J]. 今日科苑,2010,(18):247.