**1 引言**  
1.1 编写目的

为软件使用人员提供一套完善的使用方法。  
1.2 项目背景

如今信息化管理已经成为各行业的大势所趋，而医疗作为一项最基本也最重要的民生领域，更是面临着智能化、信息化管理的迫切需求。随着医疗行业的不断进步和发展，信息管理工作复杂程度也越来越高，内容也逐渐增多,依靠人工管理已经不能满足实际需求。针对传统的人工数据处理方法，存在工作量大、误码率高、修改困难、耗时耗力等问题，医院住院部信息化管理体系的建立与实施，能够加强对院区的管理，还可以提高管理工作的规范化、系统化和信息化水平，提高信息处理的效率和精确性，防止人为错误的发生，及时、准确、有效地进行信息查询、统计。

综上，为了加快建设更加智能化、信息化的住院部信息管理系统，使住院部的职能得到充分的发挥，提高服务质量与工作效率，特开展本课题的设计与研发。  
1.3 参考资料：

1. Hertin R D, Al-Sanjary O I. Performance of hospital information system in Malaysian public hospital: A review[J]. International Journal of Engineering & Technology, 2018, 7(4.11): 24-28.
2. 杨长兴, 李连捷主编. 医学计算机应用基础[M]. 北京: 中国铁道出版社, 2014.08.
3. 张彦铭.数字化背景下医院住院信息管理系统设计与实现[J]. 数码世界, 2020(09): 264-265.
4. 朱东, 吴佳迪, 蓝晨晖, 等. 基于Android的住院部管理信息系统的开发与应用[J]. 电子世界. 2014(3): 90-90.
5. 郑树军,周波. 医院管理信息系统中电子信息技术的应用[J].长江信息通信, 2022, 35(03): 176-178.
6. Hamidi M, Mahendran P, Denecke K. Towards a Digital Lean Hospital: Concept for a Digital Patient Board and Its Integration with a Hospital Information System[J]. Studies in health technology and informatics, 2019, 264: 606-610.
7. 薄春敏. 数字化医院住院管理系统的设计与实现[M]. 电子科技大学, 2014.
8. Hertin R D, Al-Sanjary O I. Performance of hospital information system in Malaysian public hospital: A review[J]. International Journal of Engineering & Technology, 2018, 7(4.11): 24-28.
9. 朱兰. 医院HIS系统住院管理子系统的分析与设计[D].云南大学,2013..
10. Shahzad K, Jianqiu Z, Zia M A, et al. Essential factors for adopting hospital information system: a case study from Pakistan[J]. International Journal of Computers and Applications, 2021, 43(1): 26-37.
11. 王冯涛. 基于SSM的地铁站智慧管理系统的设计与实现[D].北京交通大学,2021.
12. 曹华山.SSM框架在Web应用开发中的设计与实现[J].无线互联科技,2021,18(11):108-109.
13. 宋锦华.基于SSM框架的在线教育平台设计与实现[J].电脑编程技巧与维护,2022(05):27-29.
14. 徐晗. 不动产登记一窗受理数据共享系统设计与实现[D].江苏科技大学,2020.
15. Lee M H. Design and implementation of hybrid apps design based on spring MVC[J]. Journal of the Korea Convergence Society, 2019, 10(3): 395-400.
16. 张雪敏. 浅议MVC设计模式在JavaWeb中的作用[J]. 科技风, 2018, 22..
17. Di Lu, Yuping Qiu, Chun Qian, Xi Wang, Wei Tan. Design of Campus Resource Sharing Platform based on SSM Framework[J]. IOP Conference Series: Materials Science and Engineering, 2019, 490(6): 126-130.
18. Wang S, Hou P, Yun L, et al. Design and Research of Paper Management System Based on SSM Architecture[J]. Journal of Physics Conference, 2018, 1069.
19. Zhang D, Wei Z, Yang Y. Research on lightweight MVC framework based on spring MVC and mybatis[C]//2013 Sixth International Symposium on Computational Intelligence and Design. IEEE, 2013, 1: 350-353.
20. Li Y Z, Gao S, Pan J, et al. Research and application of template engine for web back-end based on MyBatis-Plus[J]. Procedia Computer Science, 2020, 166: 206-212.
21. Rajput D. Mastering Spring Boot 2.0: Build modern, cloud-native, and distributed systems using Spring Boot[M]. Packt Publishing Ltd, 2018.
22. 朱锐勋,黄亮,岳正江.基于Spring Boot的办学质量评估系统设计[J].信息技术与信息化,2022(03):134-137.
23. 章晨曦. 基于组件化的后台管理页面可视化构建系统设计与实现[D].北京邮电大学,2021.
24. 刘逸婧. 智能化功率封装数据库管理系统[D].南京邮电大学,2021.
25. Boduch A. React and React Native[M]. Packt Publishing Ltd, 2017.
26. 张蕾. 网上订餐系统的数据库设计[J]. 工业技术与职业教育, 2018, 16(1): 12-14.
27. 蔡金芷, 皮喜田, 刘洪英, 等. 个人健康管理系统数据库的设计与实现[J]. 计算机测量与控制, 2017, 25(7): 222-224.
28. 徐春丽,陈雯,施春雷,朱敏.基于计算机技术的医院人事信息化管理系统研究[J].自动化技术与应用,2022,41(05):184-186.

**2 软件概述**  
2.1 目标

本软件的目标是实现了一套信息化、自动化、现代化的住院部管理系统。将住院部的信息交给系统自动化管理，更好的规范住院部的业务流程，提升医护人员的工作效率及病人的就医体验，尽可能避免由于工作人员的疏漏导致的医疗事故，提升医院住院部的管理效益和现代化水平。  
2.2 功能

（1）完成系统功能模块的设计，实现登陆、注册、退出、修改密码功能；

（2）完成病人信息管理模块的设计，实现病人基本信息的展示功能，对病人信息的增、删、改、查功能，和病人健康信息展示功能；

（3）完成医护人员信息管理模块的设计，实现医护人员基本信息的展示与增、删、改、查功能；

（4）完成住院管理模块的设计，实现病人的出入院办理功能、医护人员分配功能和病床分配功能；

（5）完成消息模块的设计，实现系统用户之间发送消息并展示个人消息列表功能；

（6）完成医院住院部信息管理系统的前台显示与前后台交互。

2.3 性能

系统的性能指的是实现某一功能需要调用多少系统资源以及需要耗费多长时间。对于一个优秀的系统，具有良好的性能是必不可少的条件，提升系统的性能不仅可以减少资源的浪费，还能提升用户的体验，所以在本小节将对医院住院部信息管理系统进行性能需求分析，以保证系统的高效性与稳定性。

（1）响应速度

本系统要求对于一些基础的操作响应时间尽量保持在2秒以内，对于一些复杂的操作，也要将响应时间控制在5秒以内。

（2）可靠性

在系统运行期间，要保证系统不会崩溃，不会发生数据显示混乱、错误甚至丢失，不会出现数据泄漏等严重的安全性问题。

（3）易用性

要确保系统的界面简洁、结构清楚、导航清晰，各功能模块要设计的合理且易用，以便用户能够流畅的进行各种操作。

（4）可扩展性

为适应未来对各种功能的扩充，系统必须满足二次开发条件，且必须保证系统代码与体系结构之间的低耦合度。  
2.4 系统测试

系统测试指在特定情况下，通过一系列的行为来检测系统的缺陷或发现系统的故障。详细周密的系统测试才能够保证系统上线后不会出现重大故障或漏洞。本章节对将对系统的主要功能模块及业务流程进行测试，以检验系统是否存在问题，以及能否满足使用者的要求，并根据测试结果进一步对系统功能进行完善。

2.4.1 系统测试主要内容

本小节主要对几个核心业务模块进行功能测试和性能测试。其中的功能测试是检测各模块的功能是否可以正常实现，并要满足需求分析阶段提出的所有功能性需求。性能测试主要是对系统的响应时间、系统的安全性等进行测试。参与测试的功能模块包括病人信息管理模块、出入院办理模块、病床分配模块和消息管理模块。测试内容表2.1所示。

表2.1 系统测试内容表

|  |  |  |
| --- | --- | --- |
| 测试模块 | 测试类型 | 测试重点 |
| 病人信息管理模块 | 功能测试、非功能测试 | 验证各项操作的可用性 |
| 出入院办理模块 | 功能测试、非功能测试 | 验证数据库的一致性 |
| 病床分配模块 | 功能测试、非功能测试 | 验证病床申请与分配的正确性 |
| 消息管理模块 | 功能测试、非功能测试 | 验证发送消息的及时性 |

2.4.2 系统功能测试

本小节将设计不同的测试用例对各个模块进行测试，主要包括功能和性能两类测试用例。病人信息管理模块是系统的核心模块，主要对该模块的增、删、改、查等操作进行了测试，病人信息管理模块测试用例如表2.2所示。

表2.2 病人信息管理模块测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| 测试内容 | 测试步骤 | 预期结果 | 测试结果 |
| 查看病人详细信息 | 1. 打开病人信息管理页面 2. 点击查看按钮 | 弹出Drawer表单，在表单中展示病人的详细信息 | 成功弹出Drawer表单，并正确展示病人的全部信息 |
| 编辑病人信息 | 1. 打开病人信息管理页面 2. 点击编辑按钮进行编辑 | 保存后显示编辑后的信息 | 点击保存后表格自动刷新，并显示编辑后的信息 |
| 删除病人信息 | 1. 打开病人信息管理页面 2. 点击快速编辑按钮展开二级菜单，点击删除按钮 | 不再显示被删除的病人信息 | 点击保存后表格自动刷新，并不再显示被删除的病人信息 |
| 查询病人信息 | 1. 打开病人信息管理页面 2. 在查询表单中输入查询条件，点击查询按钮 | 根据查询条件展示符合的病人信息 | 点击查询后表格中展示符合查询条件的病人信息 |

出入院办理是住院部信息管理系统最主要的业务，对该模块的主要测试是在为病人办理入院和出院后，相关的数据表是否进行了及时的、正确的更新。出入院办理模块测试用例如表2.3所示。

表2.3 出入院办理模块测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| 测试内容 | 测试步骤 | 预期结果 | 测试结果 |
| 病人入院办理 | 1. 打开入院办理页面 2. 填写病人基本信息 3. 为病人分配医护人员 | 在病人信息管理页面正确的展示刚刚办理入院的病人信息，且相关医护人员负责病人数加一 | 在病人信息管理页面成功查看到刚刚办理入院的病人信息，且相关医护人员负责病人数加一 |
| 病人出院办理 | 1. 打开出院办理页面 2. 点击办理出院按钮 | 该病人的状态更新为已出院，出院时间为当前时间，病人所在的病床状态更新为未分配，相关医护人员负责病人数减一 | 该病人的状态成功更新为已出院，且出院时间为当前时间，所在的病床状态更新为未分配，相关医护人员负责病人数减一 |

病床分配模块实现病人的病床分配的功能，需要先由医生进行申请，再由管理员进行审核，对该模块的测试主要是提交的病床申请信息是否正确、审核状态能够及时更新等。病床分配模块测试用例如表2.4所示。

表2.4 病床分配模块测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| 测试内容 | 测试步骤 | 预期结果 | 测试结果 |
| 病床申请 | 打开病人信息管理页面点击病床申请按钮，并选择病床 | 在病床申请列表模块可以查看刚刚的病床申请信息，状态为待审核 | 成功查看到病床申请信息 |
| 病床审核 | 打开病床审核界面对病床申请进行审核 | 点击确定按钮后该条申请状态更新为已通过，在病人信息管理页面可以查看到病床分配信息 | 申请状态成功更新为已通过，在病人信息管理页面成功查看到病床信息 |

对消息管理模块的测试主要是验证在办理各种业务时，通知消息能否及时的发送给每个用户。消息管理模块测试用例如表2.5所示。

表2.5 消息管理模块测试用例

|  |  |  |  |
| --- | --- | --- | --- |
| 测试内容 | 测试步骤 | 预期结果 | 测试结果 |
| 分配医护人员后发送通知 | 1. 打开病人信息管理模块 2. 点击医护人员分配按钮，选择医护人员 3. 医护人员查看消息 | 医护人员收到管理员为自己分配病人的信息 | 相关医护人员成功收到病人分配通知 |
| 病床申请及审核后发送通知 | 1. 进行病床申请操作 2. 进行病床审核操作 3. 管理员与医生查看消息 | 病床申请后管理员收到申请的通知，病床审核后医生收到审核结果的通知 | 管理员成功接收到病床申请通知，且管理员审核后相关医生成功收到病床审核通知 |

2.4.3 系统性能测试

性能测试也称为非功能性测试，上一小节的功能测试验证了系统能够正确的实现需求分析阶段提出的对各个模块的需求，而性能测试主要是对系统的效率进行测试和评估，提升用户的体验。本小节测试将通过记录服务器端响应客户端发送的请求的时间来测试系统的性能。页面响应性能测试用例表如表2.6所示。

表2.6 页面响应性能测试用例表

|  |  |
| --- | --- |
| 测试项 | 性能测试 |
| 测试内容 | 查询病人信息所用时间 |
| 测试步骤 | 1. 打开病人信息管理页面 2. 输入查询条件，点击查询并开始计时 |
| 预期结果 | 从用户点击查询按钮到查看结果的总时间少于2秒 |
| 测试结果 | 用户点击查询按钮1秒钟后成功展示正确查询结果，测试通过 |

**3 运行环境**  
 本系统的运行环境是Windows操作系统，系统位数是64位，使用的浏览器是Microsoft Edge，CPU为2.0GHz以上微处理器，内存为8G，硬盘为256G。本系统采用的后端集成开发环境为[IntelliJ IDEA](https://www.jetbrains.com/zh-cn/idea-edu/" \t "_blank)，前端集成开发环境为WebStorm，开发语言使用Java和TypeScript，数据库使用的MySQL5.6。  
**4 使用说明**  
4.1 关键技术分析

4.1.1 SSM 框架集

SSM 框架集由Spring、Spring MVC、MyBatis三个开源框架整合而成，是实际开发中常用的Web项目框架。SSM的优势在于使用了MVC的三层设计思想，在软件开发过程中，SSM能与三层架构完美融合，其中Spring MVC、Spring和MyBatis分别对应了表示层(View) 、业务层（Service）和持久层（Dao）。SSM框架实现了web层的解耦，大大的减轻了代码的冗余度和程序的复杂性，并且使后续的修改变得更加简单，为我们的系统的开发和维护提供了便利。



图4.1 SSM框架和三层架构关系图

4.1.2 Spring Boot 框架

Spring Boot构建在现有Spring Framework之上，它大大降低了软件开发的难度与复杂度。所以基本上，Spring Boot 不是一个单独的框架，而是一系列预制的模块与组件，开发人员只需挑选和使用自己所需要的部分，无需其他任何开销与配置。

Spring Boot框架具有如下特性：

（1）直接使用，不需要产生代码或 XML配置，还可以根据具体需要修改缺省值；

（2）便于向外部输出各类服务，例如REST API，网络Socket，Web流等；

（3）整合了安全性政策；

（4）支持关系数据库和非关系数据库；

（5）提供了诸如嵌入式服务器、安全、指标、健康检测、外部配置等的通用非功能性特性；

（6）自动化的管理相依性，有自己的应用监视功能，能够支持多种编译器，比如 IntelliJ IDEA，NetBeans等；

（7）支持热起动的功能开发套件。

Spring Boot的这些特点为开发、测试、配置、部署和监控提供了极大的便利，大大地提升了开发的效率。

4.1.3 React 框架

React是于2013年由FaceBook发布的一款前端框架，它是用于构建用户界面的JavaScript库。组件化是React的一个重要特性，其崇尚以组件的方式去构建UI界面。在这种思想下，用户界面将会被封装成一个个组件，诸如按钮组件Button、对话框组件Dialog、日期组件Calendar等。开发人员可以将这些组件按照特定的需求组合在一起，从而形成一个具有丰富功能的交互界面。使用React完成这项工作的流程如下：首先我们编写后端代码将要展示的数据传给前端，当React组件接收到数据后，其会通过DOM节点的方式，使用HTML来展示我们的数据。

4.1.4 Ant Design设计体系

Ant Design是一套基于确定和自然的设计价值观上的模块化解决方案，让设计者和开发者能够专注于业务逻辑的开发。Ant Design主要分为“设计语言”和“组件”两部分，其中“设计语言”部分主要面向于设计师，讲了Ant Design的设计思想以及设计规范，而“组件”部分主要面向于开发人员，包括了在Web开发中最常用的一些组件的API文档。Ant Design包含了非常多的常用组件，非常适合用来开发一些后台管理系统等对UI的个性化要求不高的项目的用户界面。其使用TypeScript和JavaScript构建，并给出了一套十分完备的组件化设计标准和组件化代码规范，极大地改善了产品的开发和开发效率和质量。

4.2 数据库表结构：

1. 管理员信息表

管理员信息表用来存储管理员的信息。管理员账号密码不能为空，并且密码采用MD5加密算法进加密后将加密后进行保存，管理员的身份默认为0。管理员信息表如表4.1所示。

表4.1管理员息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| id | int | 11 | Primary key/not null | 管理员编号 |
| username | varchar | 256 | null | 姓名 |
| user\_account | varchar | 256 | not null | 登陆账号 |
| user\_password | varchar | 512 | not null | 登陆密码 |
| admin\_gender | varchar | 10 | null | 管理员性别 |
| admin\_age | int | 11 | null | 管理员年龄 |
| phone | varchar | 256 | null | 手机号 |
| user\_role | tinyint | 4 | 0 | 用户身份 |
| avatar\_url | varchar | 1024 | null | 用户头像 |

1. 科室信息表

科室表中存储了住院部所有科室的信息。科室信息表如表4.2所示。

表4.2科室信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| department\_id | int | 11 | Primary key/not null | 科室编号 |
| department\_name | varchar | 256 | not null | 科室名称 |

1. 医生信息表

医生信息表中存储了所有医生的信息，在医生注册时插入数据。医生的账号、密码以及所属科室信息在注册时必须要填写，所以不能为空。医生的密码同样使用了MD5加密算法进行了加密，医生的身份默认为1。医生信息表如表4.3所示。

表4.3医生信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| id | int | 11 | Primary key/not null | 医生编号 |
| username | varchar | 256 | null | 医生姓名 |
| user\_account | varchar | 256 | not null | 登陆账号 |
| user\_password | varchar | 256 | not null | 登陆密码 |
| doctor\_gender | varchar | 10 | null | 医生性别 |
| doctor\_age | int | 11 | null | 医生年龄 |
| phone | varchar | 256 | null | 手机号 |
| avatar\_url | varchar | 1024 | null | 用户头像 |
| department\_id | int | 11 | foreign key /not null | 所属科室编号 |
| user\_role | tinyint | 4 | 1 | 用户身份 |

1. 护士信息表

护士信息表中存储了所有护士的信息，在护士注册时插入数据。其字段的属性同医生信息表类似，护士的身份默认为2。护士信息表如表4.4所示。

表4.4护士信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| id | int | 11 | Primary key/not null | 护士编号 |
| username | varchar | 256 | null | 护士姓名 |
| user\_account | varchar | 256 | not null | 登陆账号 |
| user\_password | varchar | 256 | not null | 登陆密码 |
| nurse\_gender | varchar | 10 | null | 护士性别 |
| nurse\_age | int | 11 | null | 护士年龄 |
| phone | varchar | 256 | null | 手机号 |
| avatar\_url | varchar | 1024 | null | 用户头像 |
| department\_id | int | 11 | foreign key /not null | 所属科室编号 |
| user\_role | tinyint | 4 | 2 | 用户身份 |

1. 病床信息表

病床信息表中存储了各个科室的病床信息，在添加病床时插入数据。病床的所属科室不能为空，在添加新病床时，病床可用状态默认为1表示可用，分配状态默认为0表示未分配。病床信息表如表4.5所示。

表4.5病床信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| bed\_id | int | 11 | Primary key/not null | 病床编号 |
| department\_id | int | 11 | not null | 所属科室编号 |
| available | tinyint | 1 | 1 | 是否可用 |
| allocated | tinyint | 1 | 0 | 是否已被分配 |

1. 病人信息表

病人信息表存储了住院部中所有病人的信息，是系统的关键数据，在为病人办理入院时插入数据。入院日期由后台在插入数据的时候获取当前的日期进行插入。在插入病人信息时，其出院状态默认为0，表示未出院。病人信息表如表4.6所示。

表4.6病人信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| patient\_id | int | 11 | Primary key/not null | 病人编号 |
| patient\_name | varchar | 256 | not null | 病人姓名 |
| patient\_gender | varchar | 10 | null | 病人性别 |
| patient\_age | tinyint | 4 | null | 病人年龄 |
| bed\_id | int | 11 | foreign key /null | 病人病床号 |
| doctor\_id | int | 11 | foreign key /null | 主治医生编号 |
| nurse\_id | int | 11 | foreign key /null | 责任护士编号 |
| is\_discharged | int | 11 | 0 | 病人是否出院 |
| admission\_date | date | 0 | not null | 病人入院日期 |
| discharged\_date | date | 0 | null | 病人出院日期 |
| condition\_description | text | 0 | null | 病情描述 |
| department\_id | int | 11 | foreign key /not null | 所属科室编号 |

1. 病床分配信息表

病床分配信息表中存储了病床分配的信息，用于实现病床的申请分配业务，在医生为病人申请病床的时候插入数据。在插入申请信息时，申请状态默认为2，表示待审核。病床分配信息表如表4.7所示。

表4.7病床分配信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| patient\_id | int | 11 | Primary key/not null | 申请病人编号 |
| apply\_id | int | 11 | not null | 申请者编号 |
| bed\_id | int | 11 | not null | 申请病床编号 |
| department\_id | int | 11 | not null | 所属科室编号 |
| approve\_status | tinyint | 4 | 2 | 申请状态 |

1. 消息表

消息表中存储了所有用户的消息，在用户发送消息时插入数据。在插入消息时，其状态默认为0，表示未读。消息的发送时间在其数据插入数据库时根据当前时间自动生成。消息表如表4.8所示。

表4.8消息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| notice\_id | int | 11 | Primary key/not null | 消息编号 |
| send\_id | int | 11 | not null | 发送者编号 |
| receive\_id | int | 11 | not null | 接收者编号 |
| receive\_role | tinyint | 4 | not null | 接收者身份 |
| avatar | varchar | 1024 | null | 消息图标 |
| title | varchar | 256 | null | 消息标题 |
| content | text | 0 | null | 消息内容 |
| is\_read | tinyint | 4 | 0 | 状态 |
| data\_time | datetime | 0 | CURRENT\_TIME | 发送时间 |
| type | varchar | 32 | not null | 消息类型 |

1. 病人健康信息表

病人健康信息表存储了病人的各项生命体征信息，其数据应与病人信息表对应，在病人办理入院时插入数据。病人健康信息表如表4.9所示。

表4.9病人健康信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 数据类型 | 长度 | 约束/默认值 | 备注 |
| patient\_id | int | 11 | Primary key/not null | 病人编号 |
| patient\_name | varchar | 256 | null | 病人姓名 |
| heart\_rate | int | 11 | null | 心率 |
| pulse | int | 11 | null | 脉搏 |
| temperature | float | 0 | null | 体温 |
| blood\_pressure | float | 0 | null | 血压 |
| breathe | int | 11 | null | 呼吸 |

**5 运行说明**  
 本系统将分为用户功能模块、信息管理模块、病床管理模块、出入院办理模块、消息管理模块及个人中心模块六个主要模块，本小节将分别介绍每个模块的具体实现方法，并对各模块进行了展示。

5.1 用户功能模块实现

用户功能模块主要包括四个功能：登录功能、注册功能、修改密码功能及退出登录功能，接下来将依次对每个功能进行介绍。

（1）登录功能：登陆页面共有三个表单项，分别是账户、密码及用户身份，其中，账号密码不能为空、账号长度必须大于四位、密码长度必须大于六位等，若用户输入不满足校验，则对进行相应提示；使用Radio单选框来让用户选择自己的身份，将接收到的数据存放到封装的LoginParams类型的变量中，将其作为参数，调用login()接口向后端发送POST请求。图5.1为登录功能效果展示。

（2）注册功能：首先，只有医生和护士身份的用户可以进行注册，用户可以点击登录页面用户注册链接跳转到注册页面，注册页面表单包括账号、密码、确认密码、用户姓名及所属科室六个表单项，其中账号、密码、确认密码和用户姓名都是用ProFormText文本框进行接收，其校验逻辑同用户登录，用户身份选择也同登录功能，用户的所属科室通过ProFormSelect下拉框来选择，下拉框中展示了医院所有科室的信息。在用户提交了注册信息后，前端调用register()接口向后端发送POST请求并传递相关数据。后端接收到请求后，也需要对接收到的参数进行一系列的校验，若校验均通过，则查询数据库中是否已经有相同的账号，若没有，则调用DigestUtils.md5DigestAsHex()方法对该用户的密码进行加密，然后将用户信息插入到相应的表中，并返回注册成功的信息。用户注册功能的效果展示如图5.2所示。

图5.1 登录功能效果展示 图5.2 注册功能效果展示

（3）修改密码功能：若用户忘记了密码，则可以点击登录页面的忘记密码链接进行密码修改，可以需要用户输入自己的账号、新密码以及新密码确认，后台会查询当前账号是否存在，若存在则进行数据库表的更新操作，将密码修改为新密码，并返回修改成功信息。

（4）退出登录功能：用户头像的下拉列表中有退出登录的选项，用户点击该按钮即可实现退出登录功能，用户退出登陆后，后将session中存放的用户登录态清空，并将页面重定位到登录页。

5.2 人员信息管理模块实现

信息管理模块包括病人信息管理、医生信息管理、护士信息管理和健康信息监测四个子模块。其中只有管理员有权限访问医生、护士信息管理模块，只有护士有权限访问健康信息监测模块。接下来将依次对每个子模块的实现流程进行介绍。

（1）病人信息管理模块

病人信息管理是本系统的核心业务模块，该页面展示病人的基本信息。ProTable组件内置了查询表单及快速编辑功能，在使用时只需要按照需求对其进行修改，并将实现其请求的接口即可。首先要在api.ts文件中定义一个与后端病人实体类相对应的Patients类型，然后定义searchPatients()接口向后端发起GET请求，查询我们要展示的病人信息。在后端我们依然在Controller层中接收前端的该请求，首先，我们要去session中取到当前用户的信息，因为只有管理员可以查询所有病人的信息，而医生和护士只能查询自己所负责病人的信息。然后根据用户不同身份，设置不同的查询条件，调用Service层的list()接口查询病人信息列表，并返回给前端进行展示。病人信息管理页面的效果展示如图5.3所示。

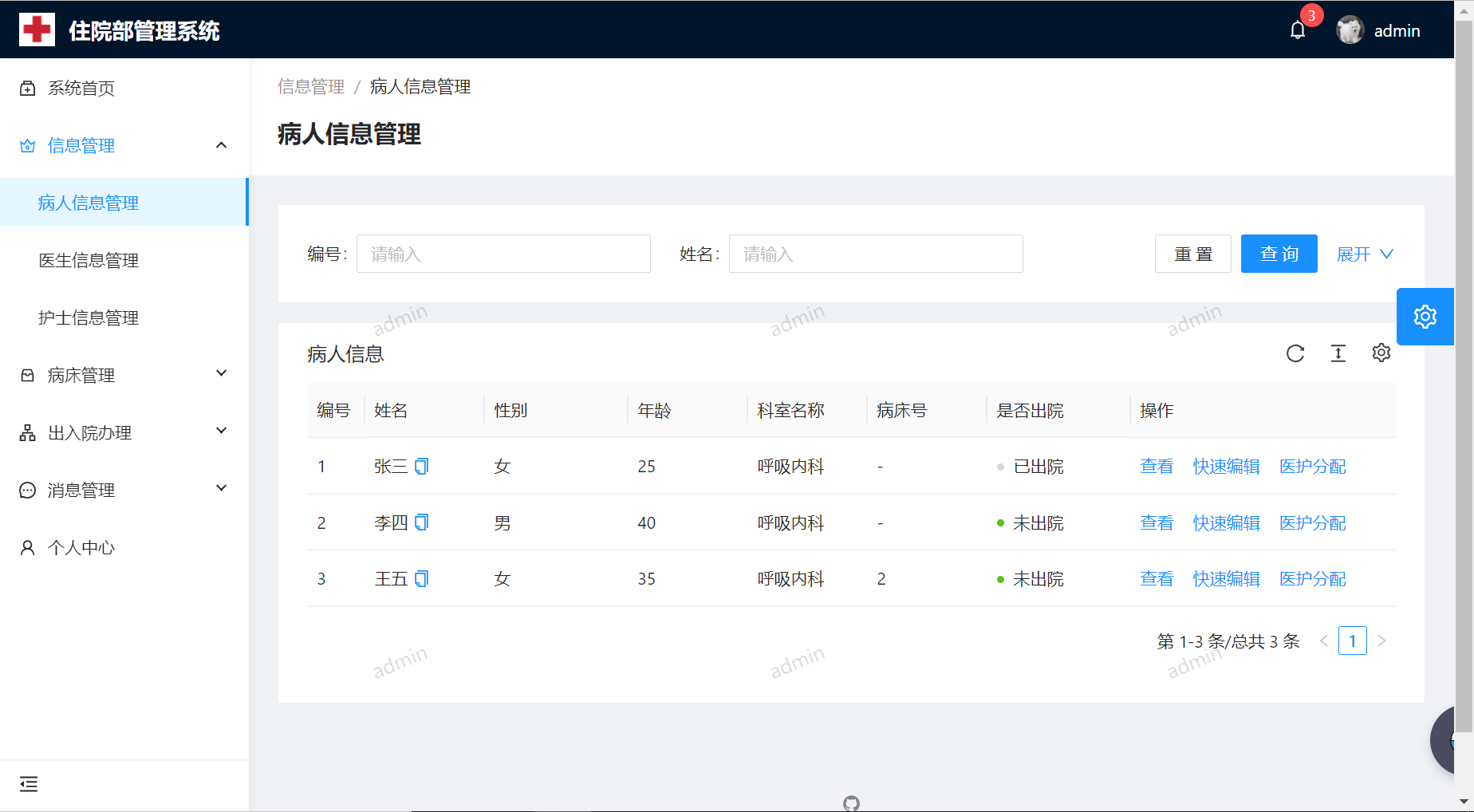


图5.3 病人信息管理效果展示

点击该页面的查看按钮可以查看病人的详细信息，使用DrawerForm表单组件对病人的详细信息进行展示。点击快速编辑按钮可以之接在表格中对病人的基本信息进行修改或删除，在完成编辑后，会调用updatePatient()接口将修改后的病人信息发送给后端，后端调用Service层的update()接口对病人信息进行更新。还以在表格上方的查询表单中输入查询条件，并可以实现多条件组合查询，查询表单中的信息会存放在formRef中，点击查询按钮会调用searchPatients()接口，并调用formRef.current.getFieldsValue()方法获取到查询表单中的信息，将其作为参数传给后端，后端需找出不为空的条件，并根据条件去Patient表中查出所需的病人信息。查询功能效果展示如图5.4所示。

在管理员端，该表格会展示一个医护人员分配的功能按钮，使用Ant Design Pro自带的Access组件实现权限控制。管理员可以点击该按钮为病人分配医护人员或对医护人员分配信息进行更改，点击该按钮会弹出一个ModelForm表单，有主治医生编号和责任护士编号两个表单项需要管理员填写，这两个表单项均采用ProFormSelect下拉框来实现，展示的是病人所属科室的所有医生和护士的编号。当点击完成按钮时，会调用后端的接口对病人信息表中的相关字段进行更新，在分配成功后会调用sendMessage()接口给相关的医护人员发送通知。医护人员分配功能效果展示如图5.5所示。



图5.4 查询功能效果展示

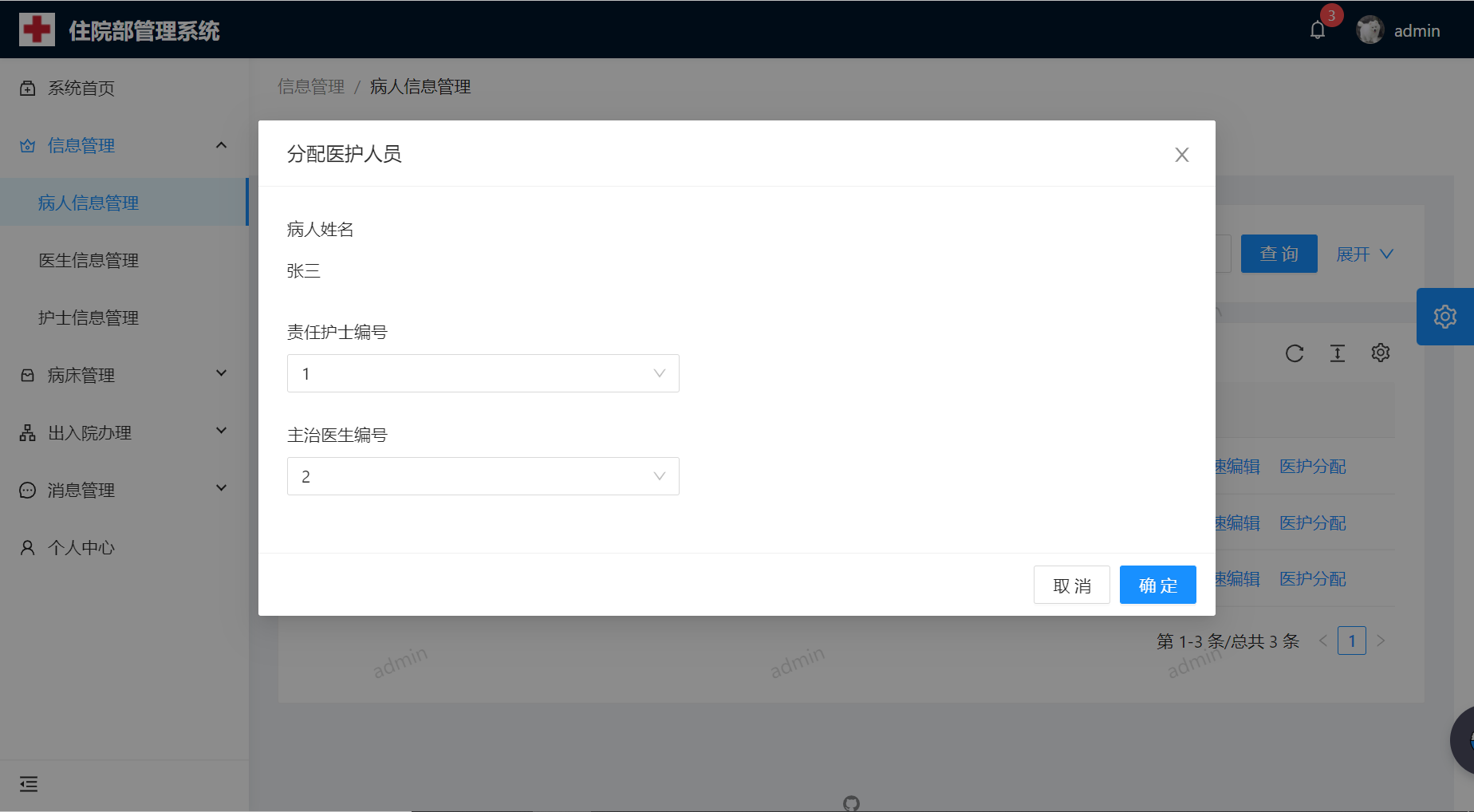


图5.5 医护人员分配效果展示

在医生端，该表格会展示一个病床申请的按钮，同样使用Access组件实现权限控制。医生可以点击该按钮为当前病人申请病床，申请病床的表单采用ModelForm组件进行展示，医生可以通过下拉框为当前病人选择本科室的可用病床，但同一位病人在同一时间内只能有一条病床申请，在申请成功后会向病床分配表中插入一条申请信息，申请状态默认为2表示审核中，然后调用sendMessage()接口给管理员发送消息通知其进行审核。病床申请功能效果展示如图5.6所示。

（2）医生信息管理模块

只有管理员有权限访问该模块，所以要对本页面进行权限配置。在access.ts文件中定义canAdmin权限，只有当前用户的身份为0时才具有该权限，然后在routes.ts文件中为该页面定义路由时，通过配置access属性来实现权限的控制。医生信息管理模块的主要功能同病人信息管理模块类似，医生信息管理模块效果展示如图5.7所示。

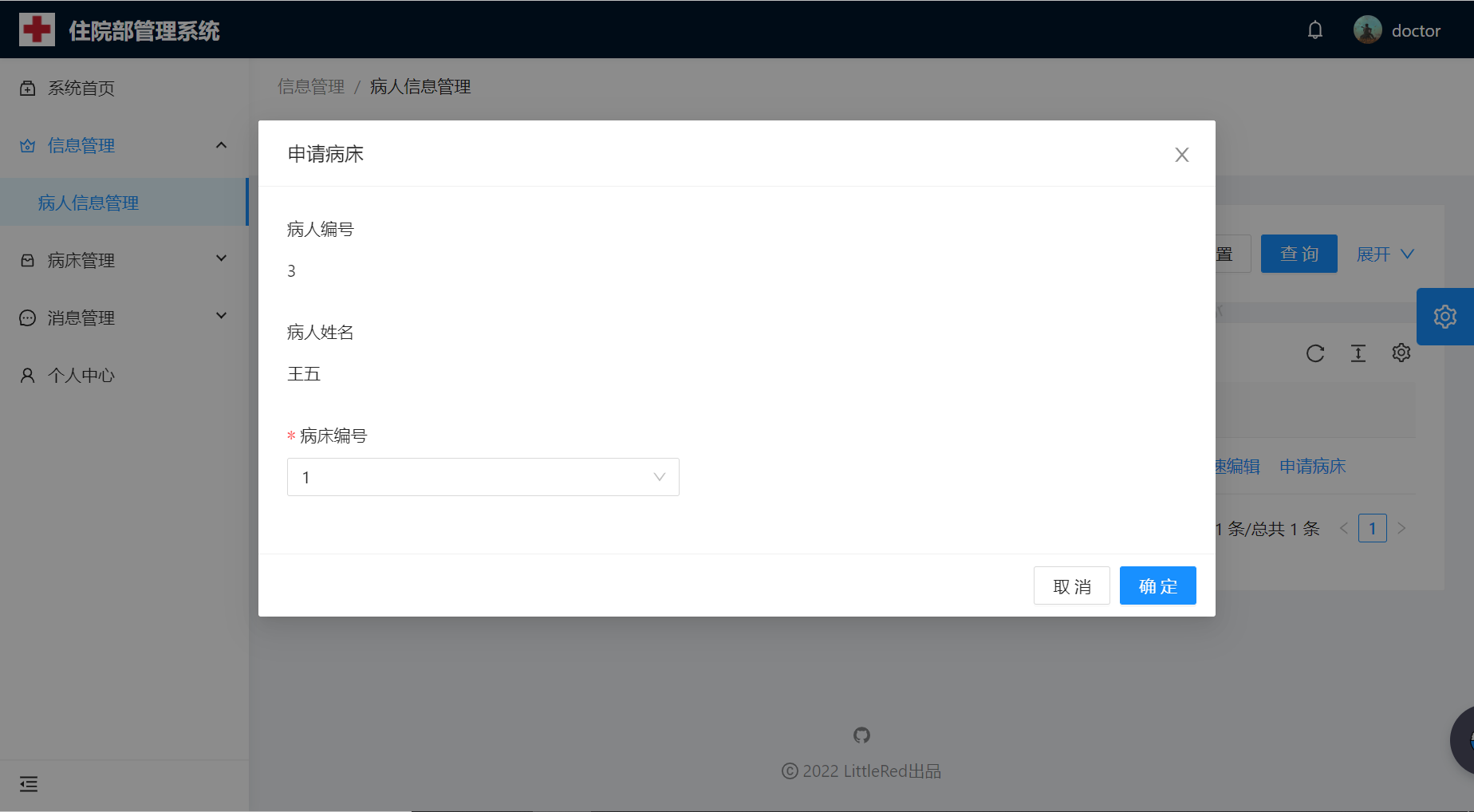


图5.6 病床申请效果展示

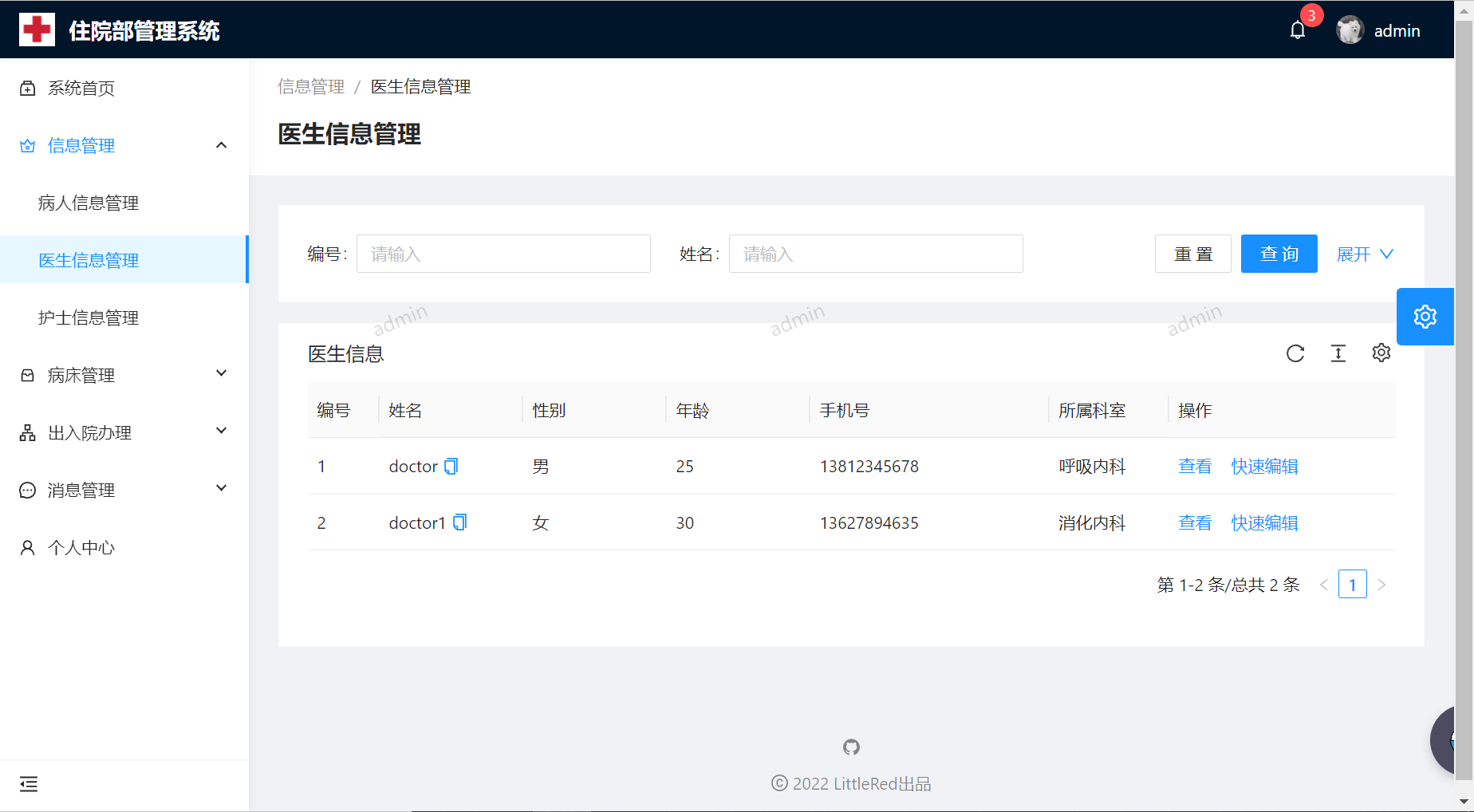


图5.7 医生信息管理模块效果展示

（3）护士信息管理模块

护士信息管理模块的主要功能及配置同医生信息管理模块类似，就不做展示。

（4）健康信息监测模块

只有护士具有访问该模块的权限，在access.ts文件中定义canNurse权限，只有当前用户的身份为2时才具有该权限。该页面使用了ProTable表格对当前护士所负责病人的各项生命体征信息进行了展示，方便护士对病人的信息进行试试监控。健康信息监测模块效果展示如图5.8所示。



图5.8 健康信息监测模块效果展示

5.3 病床管信息理模块实现

该模块包括病床信息总览、病床信息管理、病床申请列表和病床申请审批四个子模块，只有管理员和医生具有访问该页面的权限。接下来将依次对每个子模块的实现流程进行介绍。

（1）病床信息总览模块

只有管理员具有访问该模块的权限，使用canAdmin权限进行配置。该模块使用ProTable表格展示所有科室的病床信息，包括可分配病床数、已分配病床数、不可用病床数和病床总数。首先要创建病床总览信息的类型 BedOverview，将表格要展示的列与其属性一一对应，再创建查询病床总览信息的接口，向后端发起请求。在后端定义一个病床信息总览的实体类，与前端的该变量对应，接收到前端接受请求后，调用 bedService 中的接口，首先要把所有的科室列表查出来，然后遍历列表，依次去病床表中查出每个科室的可分配病床数、已分配病床数、不可用病床数、病床总数，将这些信息赋值给病床信息总览类型实体类中，然后再将该实体类插入列表，最后将列表返回给前端进行展示。

（2）病床信息管理模块

在管理员端，该页面展示了所有病床的信息；在医生端，该页面仅展示本科室的病床信息。该页面使用ProTabel表格进行展示，可进行的操作包括编辑、删除、新增和查询。首先创建病床类型Bed，属性包括：病床编号、所属科室编号、是否可用、是否分配，并将所属科室设置为枚举类型，根据不同的科室编号来展示不同的科室名称。然后调用接口请求后端查询病床信息，后端调用list()接口并设置条件，将查询出来的病床列表按照科室和病床编号递增，然后将信息返回给前端进行展示。病床信息管理模块效果展示如图5.9所示。

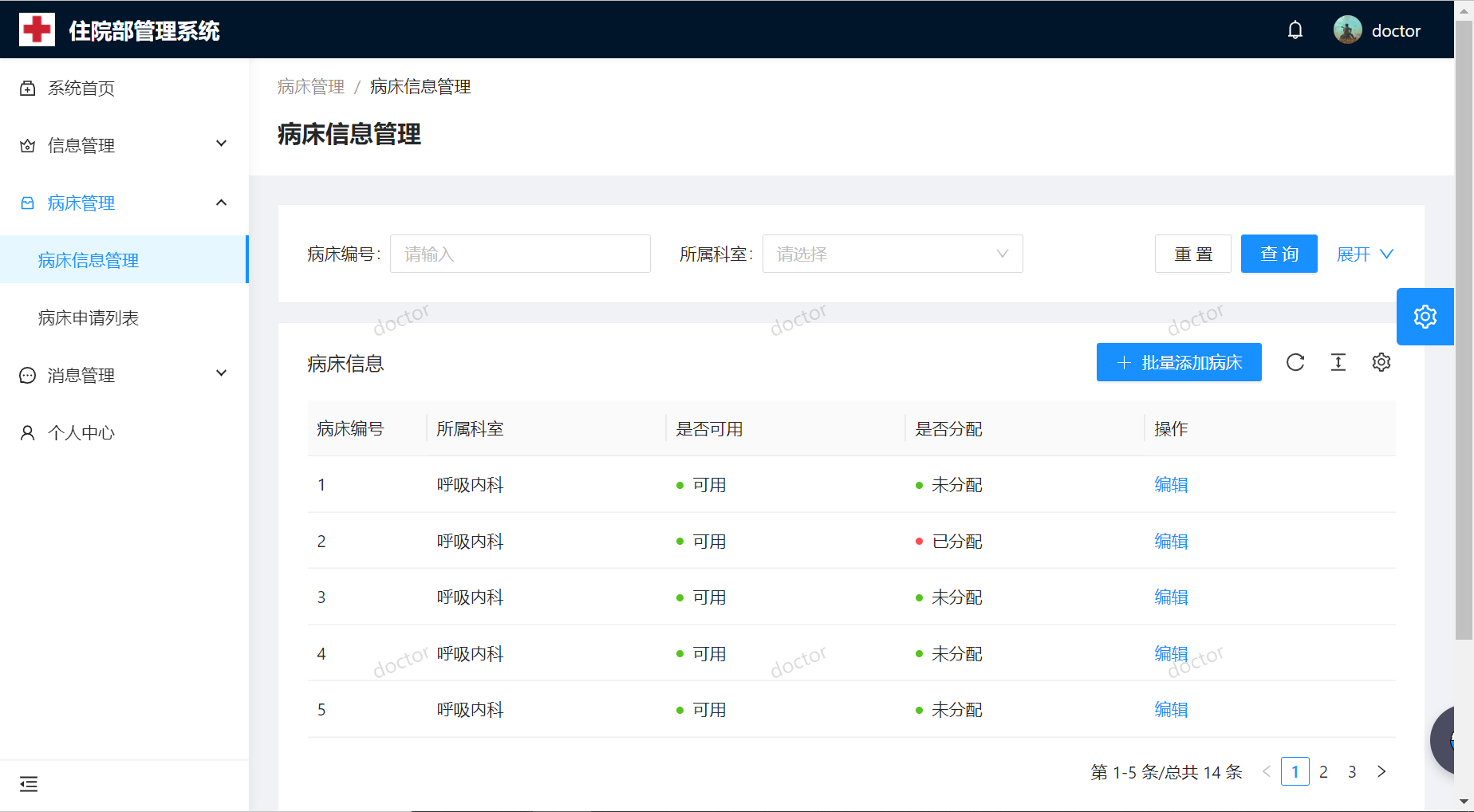


图5.9 病床信息管理效果展示

批量添加病床功能的实现流程如下：首先创建一个接收批量添加病床信息的参数AddBedParams，属性包括要添加病床的科室和添加数量，使用ModalForm表单来接收这两个参数并封装到AddBedParams实体中，然后调用接口将信息传给后台，后台接收到这AddBedParams实体的信息后，创建List<Bed>集合来存放要插入的病床信息，然后向集合中插入要添加的病床信息，调用Service层的saveBatch()接口实现批量插入。

（3）病床申请列表模块

当医生为病人申请病床后，可以在该模块查看自己提交的申请信息和审核状态，用 ProTable 来展示病床申请信息，调用接口向后端发送请求，查询病床分配表中当前医生的申请信息，并返回给前端进行展示。医生还可以修改申请或撤销申请，当医生修改申请后，会调用updateBedApply()接口向后端发送POST请求，后端会根据接收到的数据对病床分配表进行更新，并将审核状态重新置为待审核。在操作完成后会调用sendMessage()接口对管理员进行通知。病床申请列表效果展示如图5.10所示。

（4）病床申请审批模块

只有管理员具有访问该模块的权限。在该模块管理员可以查看所以医生的病床申请信息，使用ProTable表格对信息进行展示。管理员可以对病床申请信息病进行审核，选择同意或拒绝，在管理员选择时会进行判断，若当前申请已经审核过，则会提示审核失败。在选择完成后会调用approveBedApply()接口请求后端，在后端对病床分配表中的该



图5.10 病床申请列表效果展示

条申请状态进行更新，若同意申请则还需要更新病人表，为其分配该病床。在审核完成后，会将审核结果发送给提交审核的医生。病床申请审批的效果展示如图5.11所示。



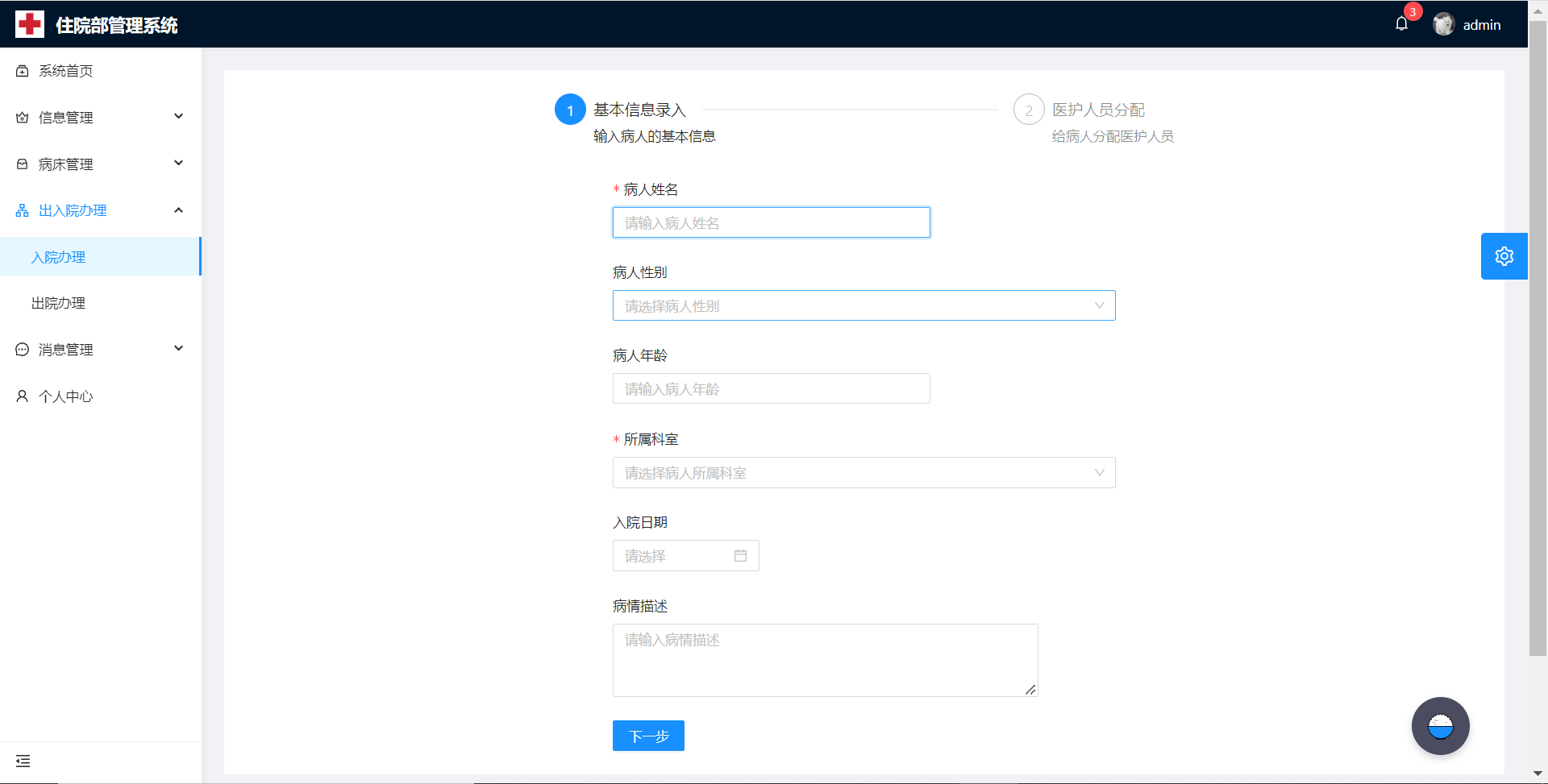
图5.11 病床申请审批效果展示

5.4 出入院办理模块实现

该模块具有出院办理和入院办理两个子模块，并且只有管理员具有访问该模块的权限。接下来将依次对每个子模块的实现流程进行介绍。

（1）入院办理模块

入院办理模块使用StepsForm分布表单实现。表单的第一步需要填写病人姓名、性别、年龄、所属科室等基本信息，其中病人的姓名与所属科室是必填项，需将rules属性中的required设为true。在填写完病人基本信息后进入下一步，为病人分配主治医生和责任护士，需根据第一步选择的科室信息，查询当前科室的所有医生和护士的信息，并使用ProFormSelect下拉列表进行展示。在所有信息都填写完毕后，点击提交按钮会调用handleAdmission()接口向后端发送POST请求，并将表单中的信息作为参数传给后端，后端在接收到病人信息后，要进行校验，包括病人姓名不能包含非法字符、病人年龄应大于零等，在校验通过后，调用Service层的save()接口将病人信息插入Patient表中。入院办理完成后会自动跳转到病人信息管理页面，即可查看到刚刚办理入院的病人信息。入院办理模块效果展示如图5.12所示。



第一步表单（a）



第二步表单（b）

图5.12 入院办理模块效果展示

（2）出院办理模块

出院办理模块使用ProTable表格展示病人出入院有关的信息，管理员可以点击出院办理按钮为当前病人办理出院，点击按钮后会调用handleDischarge()接口向后端发送POST请求，对当前病人的信息进行修改，将其是否出院的状态改为已出院，并调用LocalDate.now()方法将出院日期设置为当前的日期，然后调用update()接口对数据库进行更新。出院办理模块的效果展示如图5.13所示。

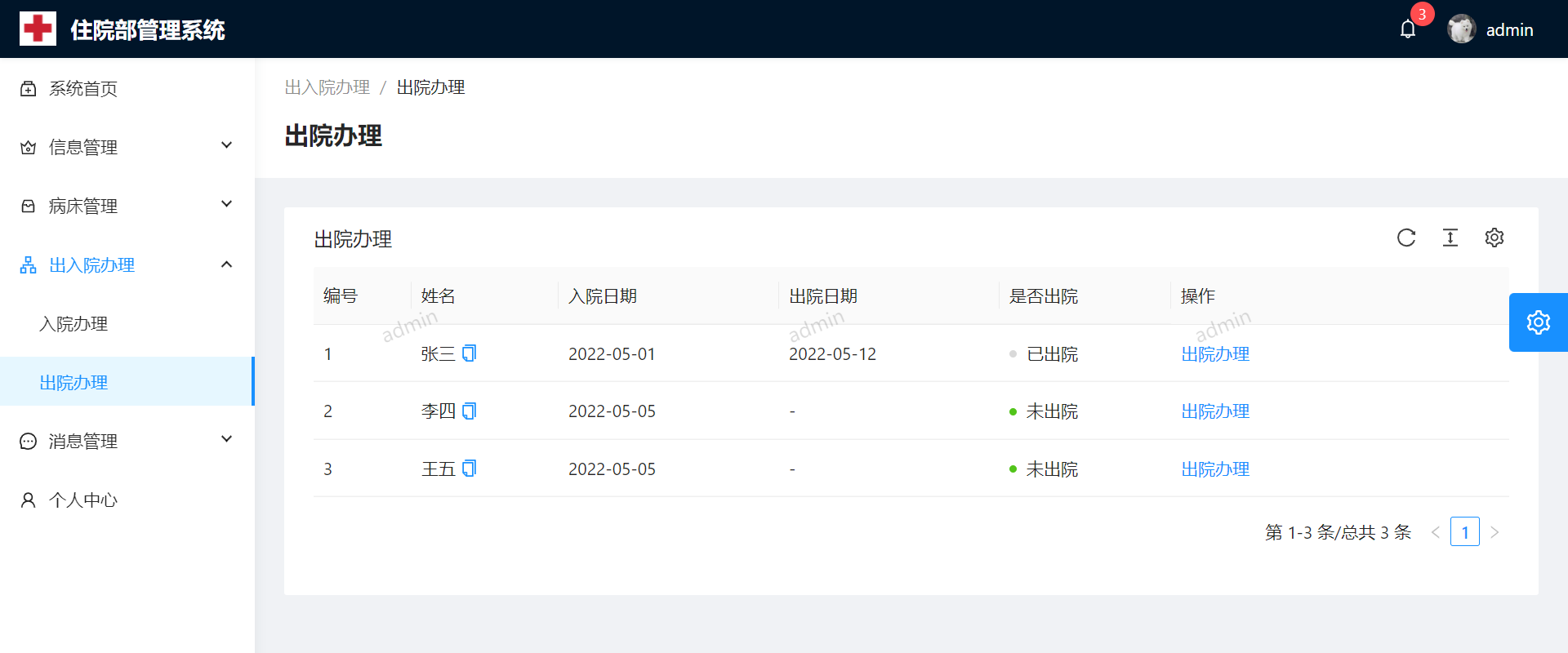


图5.13 出院办理模块效果展示

5.5 消息管理模块实现

该模块包括消息列表和发布公告两个子模块，其中只有管理员有权限访问发布公告模块。接下来将对每个子模块的实现流程进行介绍。

（1）消息列表模块

And Design Pro内置了一个消息展示的组件，该组件在系统的顶部导航栏中以一个消息的图标进行展示，该图标以红色上角标的形式展示当前用户未读消息的数量。当用户点击该图标后，可以查看收到消息的简略信息，包括消息和通知两种类型，可以进行切换查看。每条消息有标题、内容、图标三个属性，有和暗两种状态，表示是否已读，点击某条消息，接口就会将其置为灰色，并调用setIsRead()将其状态置为已读。还有清空消息和查看更多两个功能按钮，点击清空按钮，会调用clearReadState()接口将所有的消息都置为已读，点击查看更多按钮则会跳转到消息列表页面。消息图标的效果展示如图5.14所示。

在消息列表页面用户可以查看自己收到的消息，使用ProTable表格对消息进行展示。该页面具有查看和删除两个功能，点击查看按钮，会弹出一个ModalForm表单，在表单中嵌入一个ProCard卡片组件来展示消息的详细内容，并且在查看了某条消息后，会调用setIsRead()接口，将消息的状态改为已读。点击删除按钮会调用deleteNotice()接口向后端发送DELETE请求，将消息表中的该条消息删除。消息列表效果展示如图5.15所示。



图5.14 消息图标效果展示

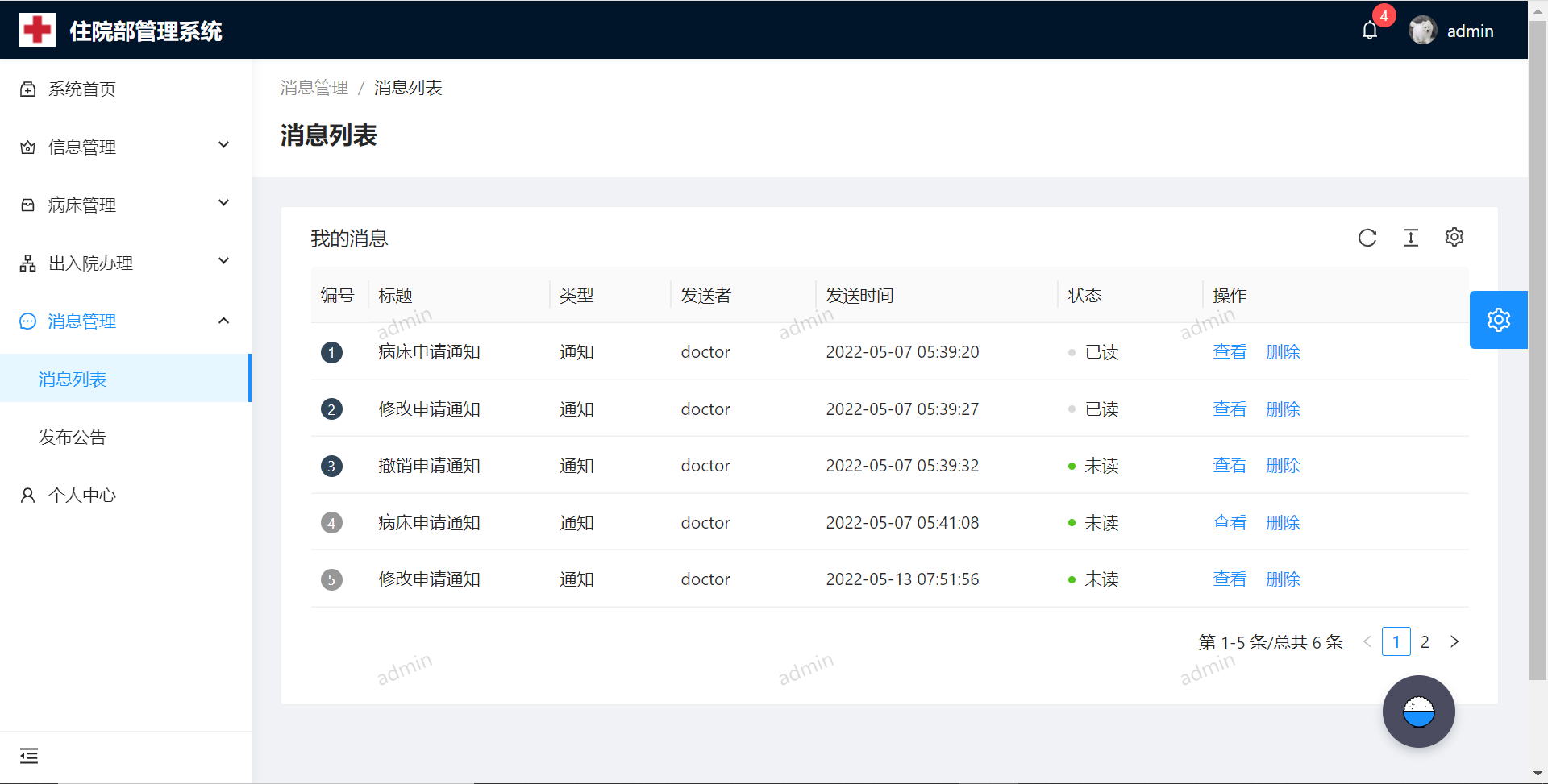


图5.15消息列表模块效果展示

（2）发布公告模块

该模块使用ProForm表单实现，有公告标题和公告内容两个表单项，其中公告标题使用ProFormText文本框进行接收，公告内容使用ProFormTextArea文本域进行接收。点击发布后，调用sendNotice()接口，将消息发送给后端，然后后端给每一位用户都添加一条公告消息。发布公告模块效果展示如图5.16所示。

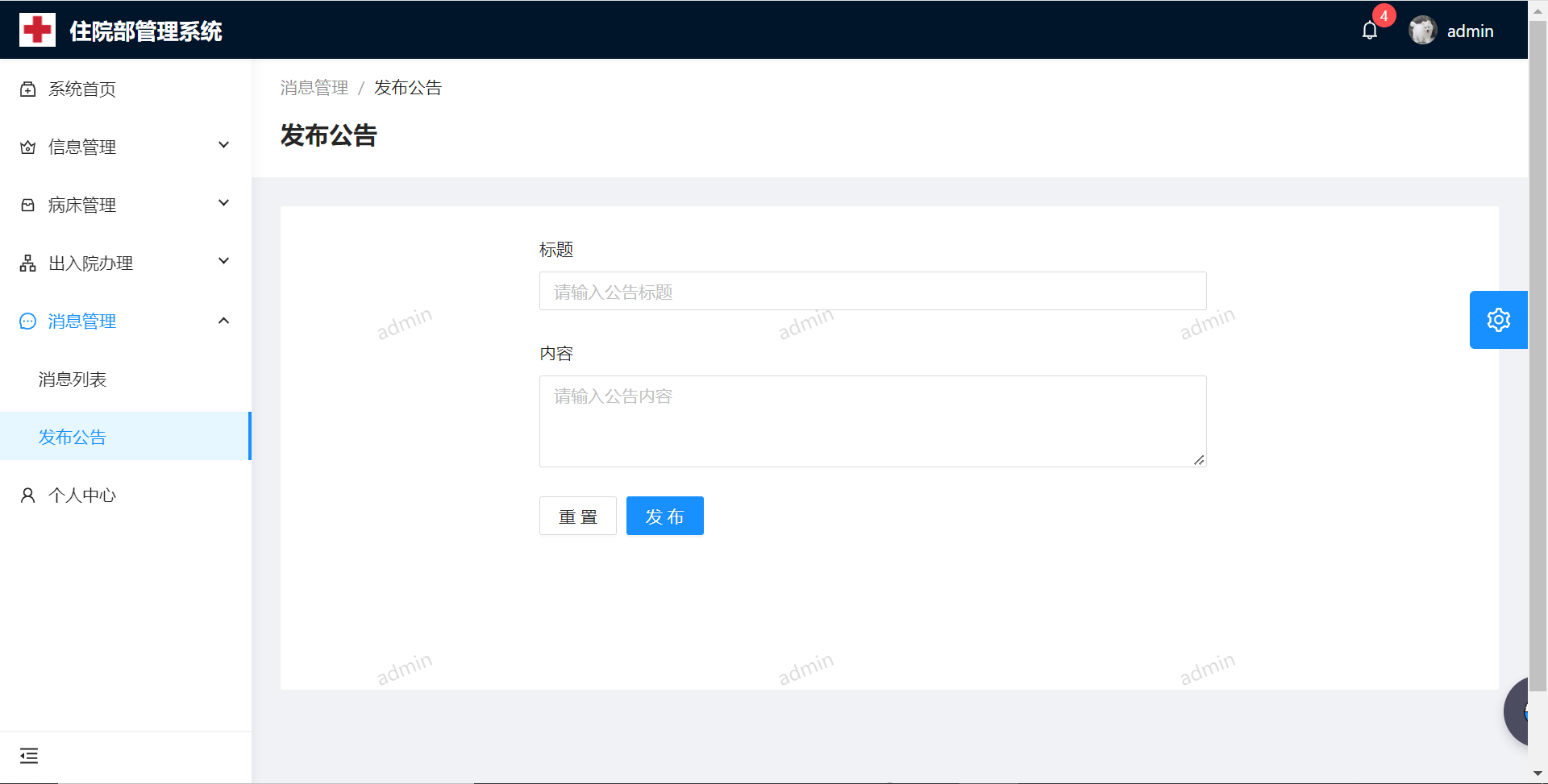


图5.16发布公告模块效果展示

5.6 个人中心模块实现

该模块使用ProCard卡片内嵌ProForm表单进行展示。首先调用getCurrentUser()接口向后端发送GET请求，后端从session中查出当前用户的信息，并对信息进行重新封装后传给前端，前端进行展示。用户还可以对个人信息进行修改，修改完成后点击保存按钮则会调用updateCurrentUser()接口将修改后的个人信息传给后端，后端先根据用户的身份来确定用户信息所属的数据表，然后根据用户的编号调用update()接口对信息进行更新。个人中心模块效果展示如图5.17所示。

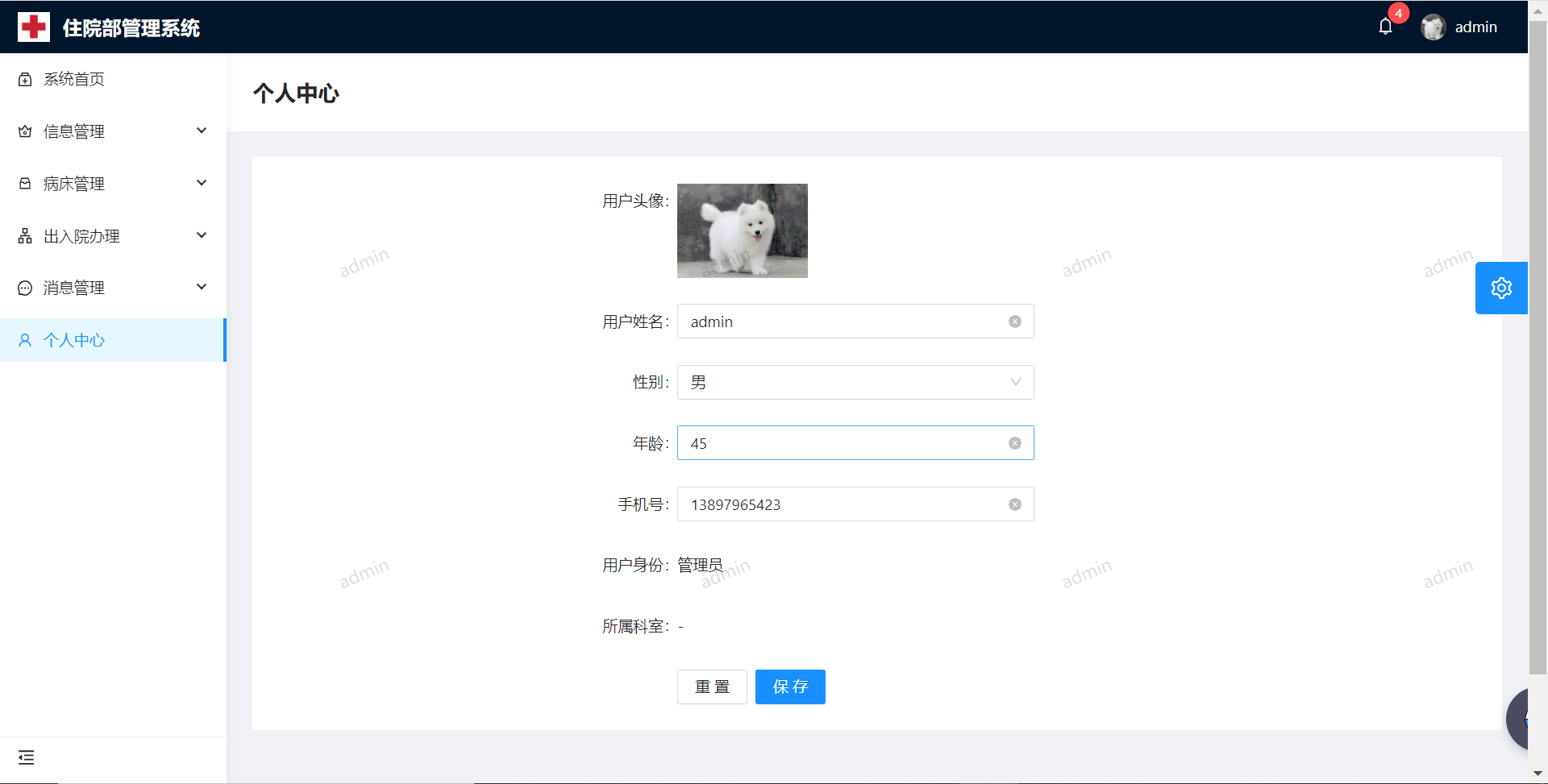


图5.17 个人中心模块效果展示