**北京邮电大学课程设计报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **课程设计**  **名称** | **面向对象程序设计实践课（Java）** | | **学 院** | **计算机** | **指导教师** | **吴起凡** |
| **班 级** | **班内序号** | **学 号** | | **学生姓名** | **成绩** | |
| 305 |  | 2022211637 | | 廖轩毅 |  | |
| 305 |  | 2022211683 | | 张晨阳 |  | |
| 307 |  | 2022211216 | | 邱梓涵 |  | |
| 307 |  | 2022211215 | | 魏征 |  | |
| 307 |  | 2022211205 | | 李梓健 |  | |
| **课**  **程**  **设**  **计**  **内**  **容** | 本课程设计旨在实现一个专家挂号系统，建设一个用数据库支持数据管理的网站，供医生患者挂号、发布出诊信息等。  本项目基于MySQL5.8，spring boot框架和Vue.js3.0,实现了三种账号的管理，患者的挂号、修改信息功能，医生的发布、查看、修改信息等功能，以及管理员的审批、修改信息等功能,为解决该医院的需求做出了相应的开发。  在本次课程设计中，我们小组采用gitee进行项目的版本控制，便于我们合作完成代码和版本迭代；除此之外，本组采用前后端分离的开发方法，前端和后端独立实现，最后统一交互接口，从而达到提高开发效率的目的。  项目分工如下：  数据库搭建：廖轩毅22%、魏征26%、李梓健26%、邱梓涵26%  后端开发：李梓健33%、邱梓涵33%、魏征33%  前端界面开发：张晨阳70%、廖轩毅30%  前后端连接：张晨阳50%、邱梓涵25%、魏征25%  报告撰写：廖轩毅70%、邱梓涵10%、张晨阳10%、魏征10% | | | | | |
| **课**  **程**  **设**  **计**  **成**  **绩**  **评**  **定** | **评语**:  **成绩**:  指导教师签名：  年 月 日 | | | | | |

注：评语要体现每个学生的工作情况，可以加页。

**目录**

[1 项目概述 1](#_Toc170893628)

[1.1 项目背景 1](#_Toc170893629)

[1.2 项目分析 1](#_Toc170893630)

[2 开发环境与语言 4](#_Toc170893631)

[2.1 开发环境与工具 4](#_Toc170893632)

[2.2 编程语言选择 4](#_Toc170893633)

[3 需求分析 5](#_Toc170893634)

[3.1 患者（普通用户） 5](#_Toc170893635)

[3.2 专家 12](#_Toc170893636)

[3.3 管理员 19](#_Toc170893637)

[4 面向对象分析 28](#_Toc170893638)

[4.1 Admin类 28](#_Toc170893639)

[4.2 Doctor类 29](#_Toc170893640)

[4.3 Patient类 30](#_Toc170893641)

[4.4 Reginfo类 31](#_Toc170893642)

[4.5 Atteninfo类 32](#_Toc170893643)

[4.6 Register类 33](#_Toc170893644)

[4.7 Changes类 34](#_Toc170893645)

[5 面向对象设计 35](#_Toc170893646)

[6 面向对象测试 36](#_Toc170893647)

[6.1 系统首页 36](#_Toc170893648)

[6.2 患者系统 37](#_Toc170893649)

[6.3 专家系统 40](#_Toc170893650)

[6.4 管理员系统 41](#_Toc170893651)

[7 需求新增 45](#_Toc170893652)

[7.1 患者违约信息 45](#_Toc170893653)

[7.2 号源拍卖功能 45](#_Toc170893654)

[7.3 用户信息新增 46](#_Toc170893655)

[7.4 相似挂号推荐 46](#_Toc170893656)

[8 项目总结 47](#_Toc170893657)

[8.1 总结体会 47](#_Toc170893658)

[8.2 后续改进 47](#_Toc170893659)

[9 分工情况 48](#_Toc170893660)

1 项目概述

1.1 项目背景

某医院委托我们开发一个专家挂号系统。要求建设一个网站供医生患者挂号，后台数据用数据库支持数据管理。估计大约患者，医生，每天平均挂号数量个，访问量，且集中在早上左右，因为医院此时开放新一天的预约号。

1.2 项目分析

1.2.1 登录系统

登录系统有注册功能和登录功能。

在注册功能中，用户需要选择注册的账号类型（普通用户、医生、管理员），并输入身份证号、用户名、密码，并再次输入确认密码。两次输入的密码不同将无法注册。若该用户名已注册该账号类型也无法注册。注册成功后将交由管理员审核，审核通过后即可登录。

在登录功能中，用户需要选择登录的账号类型（普通用户、医生、管理员），并输入用户名及对应密码。该账号类型的用户名不存在或密码不正确将无法登录。登录成功后将根据登录的账号类型进入不同的系统（管理员系统、专家系统、患者系统）。

1.2.2 管理员系统

管理员系统的功能有修改密码、修改自身信息、修改他人信息、新用户审批、出诊信息审批、退出登录。

在修改密码功能中，用户需要输入该用户名对应的旧密码和希望修改为的新密码。旧密码不正确或新密码与旧密码相同将无法修改。修改成功后点击返回首页按钮即可返回管理员系统首页。

在修改自身信息功能中，用户可以修改自身信息。修改完成后，点击保存修改按钮即可保存修改。点击返回按钮即可返回管理员系统首页。

在修改他人信息功能中，用户可以修改其他用户信息。系统将为用户列出所有用户的用户名及身份证号。用户可以选择一项进行修改，点击修改账号信息按钮即可跳转到修改用户信息页面，修改完信息后点击保存修改按钮即可保存修改，点击返回按钮即可返回修改他人信息页面。用户也可以选择一项进行删除账号操作，点击删除账号按钮即可完成删除。点击返回按钮即可返回管理员系统首页。

在新用户审批功能中，用户可以审批新用户。系统将为用户列出所有新用户的用户名及身份证号。用户可以选择一项审批通过或审批不通过，通过点击相应按钮完成相应操作。审批通过的用户将能在登录系统通过审批通过的用户名及相应密码完成登录；审批不通过的用户将无法在登录系统登录，需要重新提交注册请求。点击返回按钮即可返回管理员系统首页。

在出诊信息审批功能中，用户可以审批出诊信息。系统将为用户列出所有待审核的出诊信息，包括出诊信息的日期、时间、医院、科室、医生和总挂号数量。用户可以选择一项审批通过或审批不通过，通过点击相应按钮完成相应操作。审批通过的出诊信息将展示在专家系统的查看出诊信息功能中；审批不通过的出诊信息将被删除。

在退出登录功能中，用户点击退出登录按钮即可返回登录系统首页。

1.2.3 专家系统

专家系统的功能有查看并修改个人信息、发布出诊信息、查看出诊信息、查看出诊安排、退出登录。

点击个人信息按钮即可查看并修改用户的个人信息。修改完成后，点击保存修改按钮即可保存修改。点击返回按钮即可返回专家系统首页。

在发布出诊信息功能中，用户可以发布自己的出诊信息。用户需要选择该出诊信息的日期、具体时间及挂号数量，然后点击发布按钮即可发布出诊信息。

在查看出诊信息功能中，用户需要筛选日期，系统将为用户列出符合日期的该用户所有的出诊信息，每条出诊信息包括该出诊信息的日期、具体时间、总挂号数量及已被挂号量。用户也可以点击出诊信息右侧的删除按钮删除该出诊信息。

在退出登录功能中，用户点击退出登录按钮即可返回登录系统首页。

1.2.4 患者系统

患者系统的功能有查看并修改个人信息、预约挂号、查看挂号记录、退出登录。

点击个人信息按钮即可查看并修改用户的个人信息。修改完成后，点击保存修改按钮即可保存修改。点击返回按钮即可返回患者系统首页。

在预约挂号功能中，用户需要选择想要查询挂号信息的日期、所属医院、对应科室、对应医生（可以选择全部医生），点击查询按钮即可查询到符合条件的挂号信息。点击选择挂号按钮并确认支付即可完成挂号。

在查看挂号记录功能中，系统将为用户列出该用户所有的挂号记录，每条记录包括该挂号记录的日期、医院、科室和医生。点击返回按钮即可返回患者系统首页。

在退出登录功能中，用户点击退出登录按钮即可返回登录系统首页。

2 开发环境与语言

2.1 开发环境与工具

2.1.1 操作系统

项目开发与运行环境为Windows11 x64系统。此操作系统具有良好的用户界面和强大的硬件兼容性，能有效支持开发工作。

2.1.2 集成开发工具

使用IDEA 2023.3.4作为主要的集成开发环境。IDEA提供了丰富的插件和工具支持，包括代码自动补全、重构、调试等功能，提高了开发效率和代码质量。

2.1.3 SpringBoot框架

SpringBoot是一个简化Spring应用开发的框架，通过提供一系列的默认配置，帮助开发者快速创建独立、生产级别的Spring应用程序。

2.1.4 MySQL5.8

MySQL 5.8作为项目的关系型数据库管理系统，提供了高性能和高可靠性的数据库服务，支持复杂的查询和事务处理，适合处理大量数据的应用场景。

2.2 编程语言选择

2.2.1 JDK1.8

项目选择使用JDK1.8作为Java开发工具包。这是因为JDK1.8与SpringBoot框架高度兼容，能够充分利用SpringBoot的特性和功能。Java语言以其跨平台性、强类型安全性和丰富的类库，广泛应用于企业级开发中。

2.2.2 Vue3.0

Vue 3.0 在本项目中扮演的角色是构建用户界面（UI）的前端 JavaScript 框架。Vue 3.0提供了一个强大、灵活且高效的前端开发解决方案，用于构建交互性强、用户友好的 Web 应用。

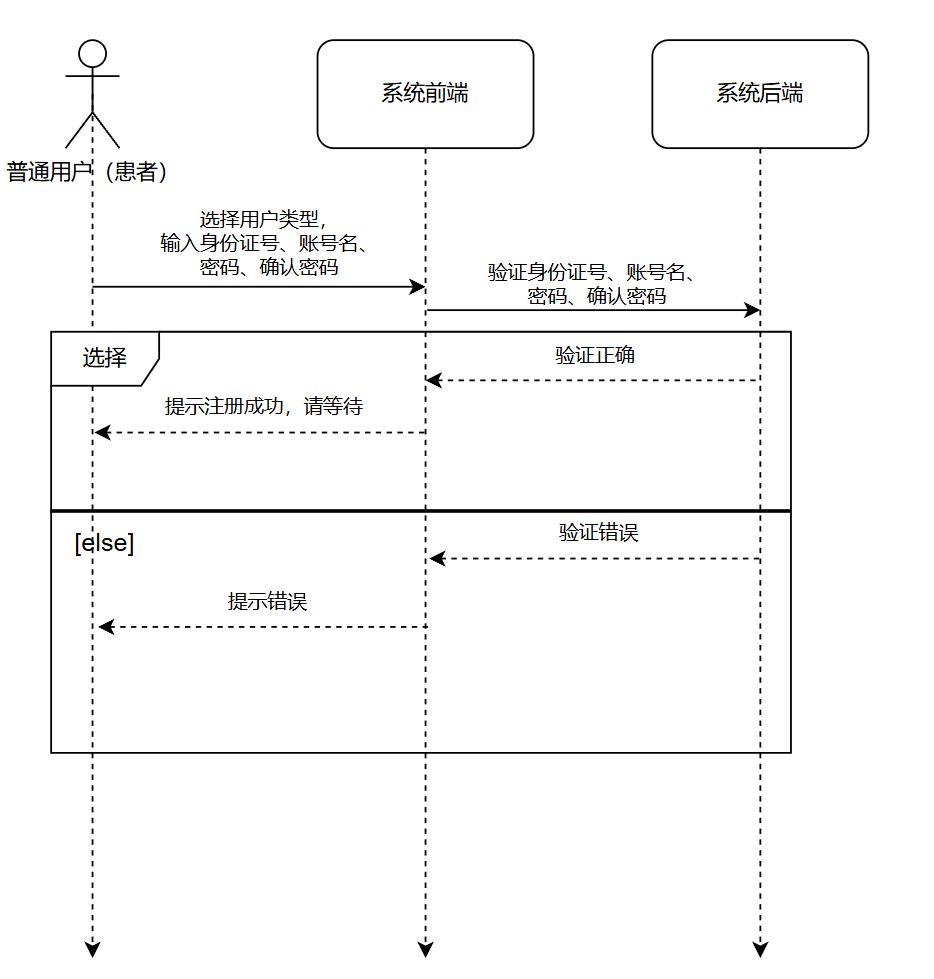
3 需求分析

3.1 患者（普通用户）



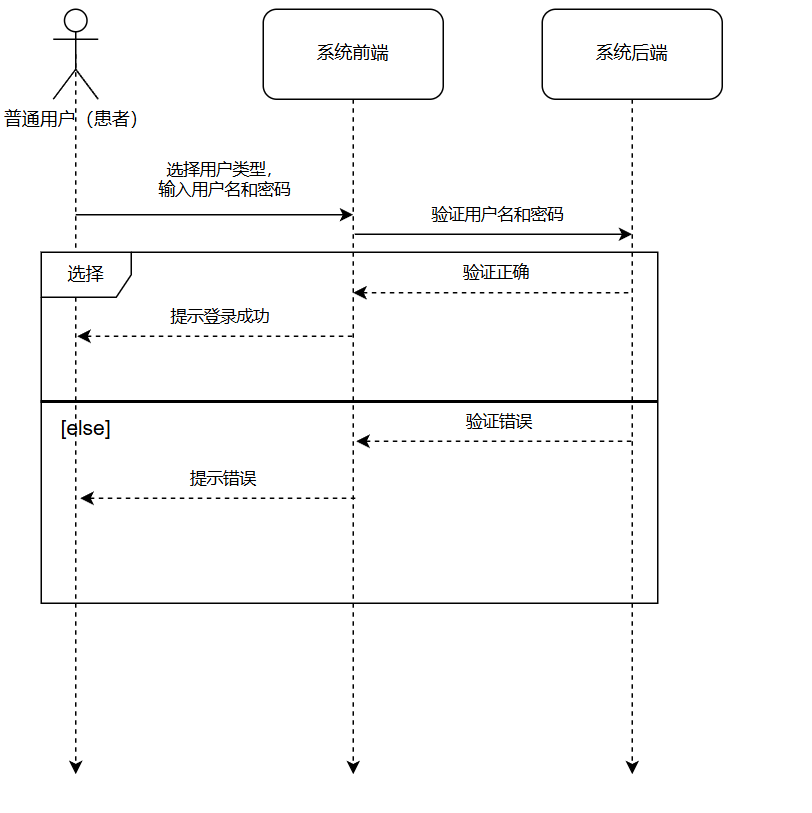
患者可以注册、登录账号，登录账号后可以查看个人信息、查看挂号记录、预约挂号、退出登录。

3.1.1 注册账号



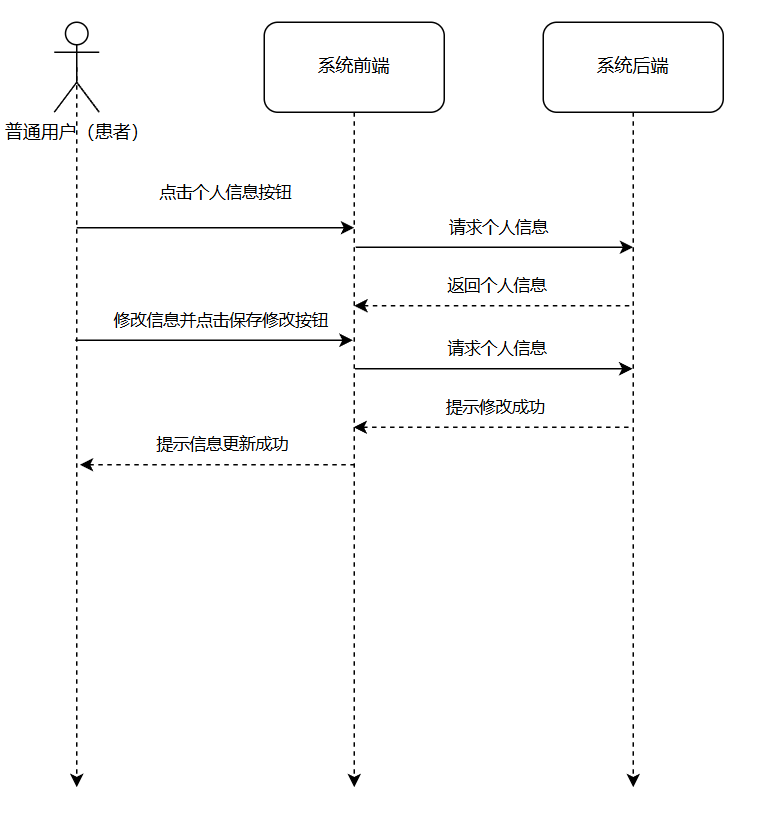
用户在前端选择用户类型，并输入身份证号、账号名、密码及确认密码，点击注册按钮后，系统前端向系统后端发送数据。后端进行数据验证，验证成功则将注册数据存入存放注册数据的数据库中，并告诉前端验证正确，前端提示用户注册成功，请等待；验证失败则告诉前端验证错误，前端提示用户错误信息。

3.1.2 登录账号



用户在前端选择用户类型，并输入账号名和密码，点击登录按钮后，系统前端向系统后端发送数据。后端进行数据验证，验证成功则告诉前端验证正确，前端提示用户登录成功；验证失败则告诉前端验证错误，前端提示用户错误信息。

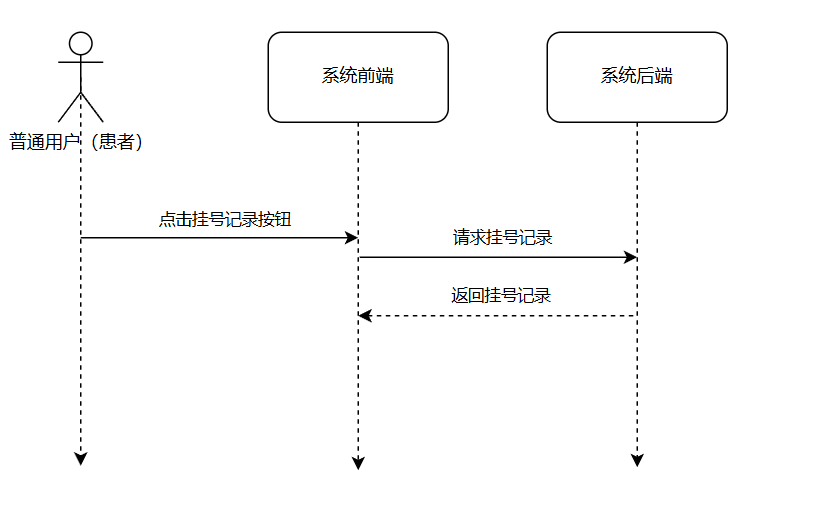
3.1.3 查看个人信息



用户点击个人信息按钮，系统前端跳转到个人信息主页，并向系统后端请求该用户的个人信息，系统后端返回该用户的个人信息并展示在系统前端中。

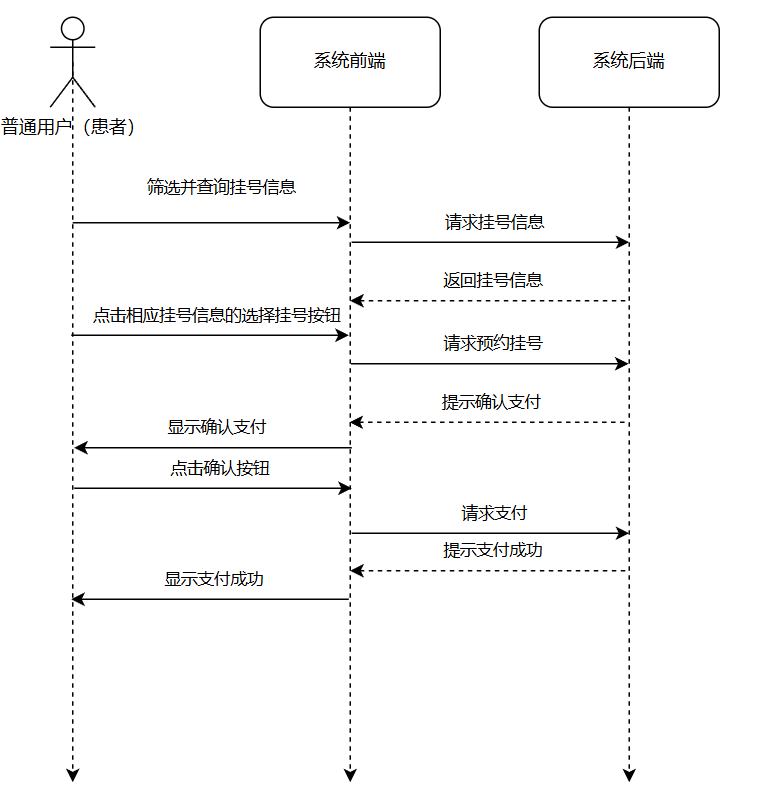
用户修改个人信息并点击保存修改按钮，系统前端将修改后的个人信息发送至后端，后端更新数据库并向前端提示修改成功，前端提示用户信息更新成功。

3.1.4 查看挂号记录



用户点击挂号记录按钮，系统前端跳转到挂号记录页面，并向系统后端请求该用户的挂号记录，系统后端返回该用户的挂号记录并展示在系统前端中。

3.1.5 预约挂号

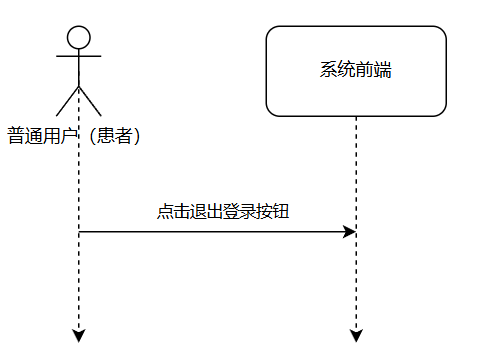


用户筛选挂号信息并点击查询按钮后，系统前端向系统后端发送筛选要求，后端向前端发送符合要求的挂号信息，前端向用户展示符合要求的挂号信息。

用户点击相应挂号信息的选择挂号按钮后，系统前端向系统后端发送预约挂号请求，后端提示前端确认支付，系统前向用户显示确认支付。

用户点击确认支付按钮后，系统前端向系统后端发送支付请求，系统后端将该挂号记录添加至用户存放挂号记录的数据库中并提示系统前端支付成功，系统前端向用户显示支付成功。

3.1.6 退出登录



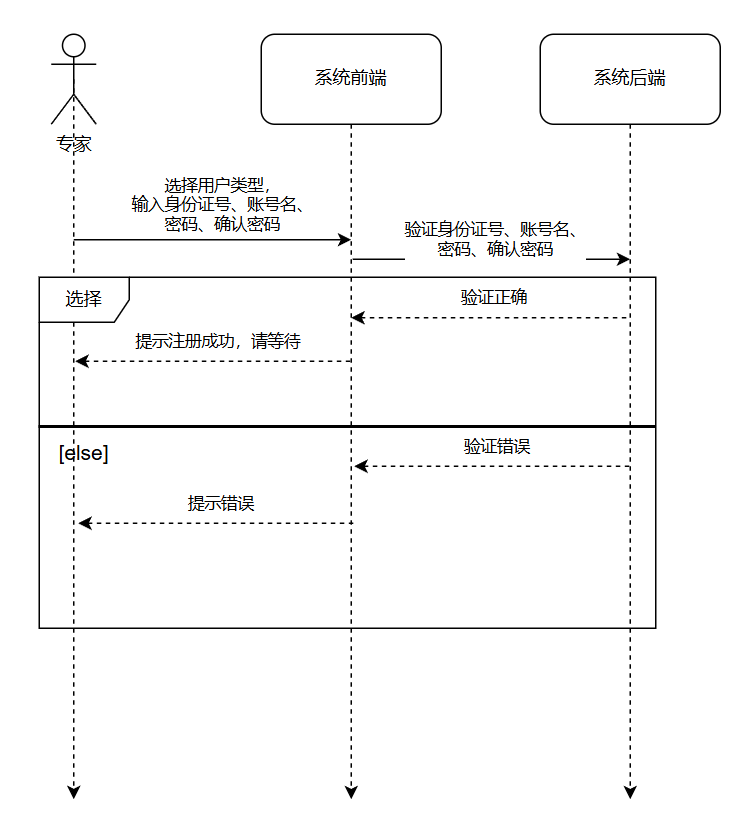
用户点击退出登录按钮后，系统前端回到系统登录首页。

3.2 专家



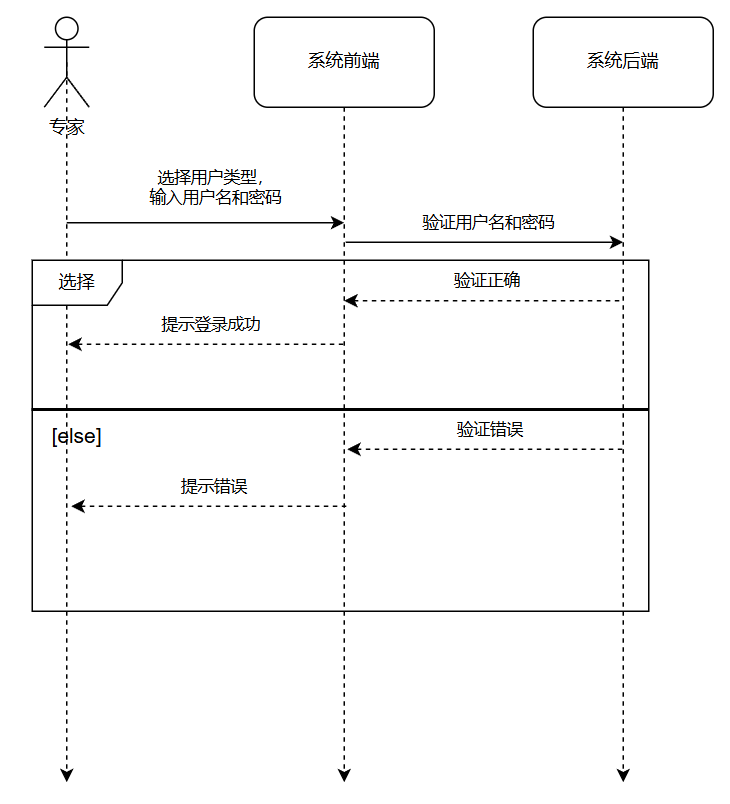
专家可以注册、登录账号，登录账号后可以查看个人信息、发布出诊信息、查看出诊信息、退出登录。

3.2.1 注册账号



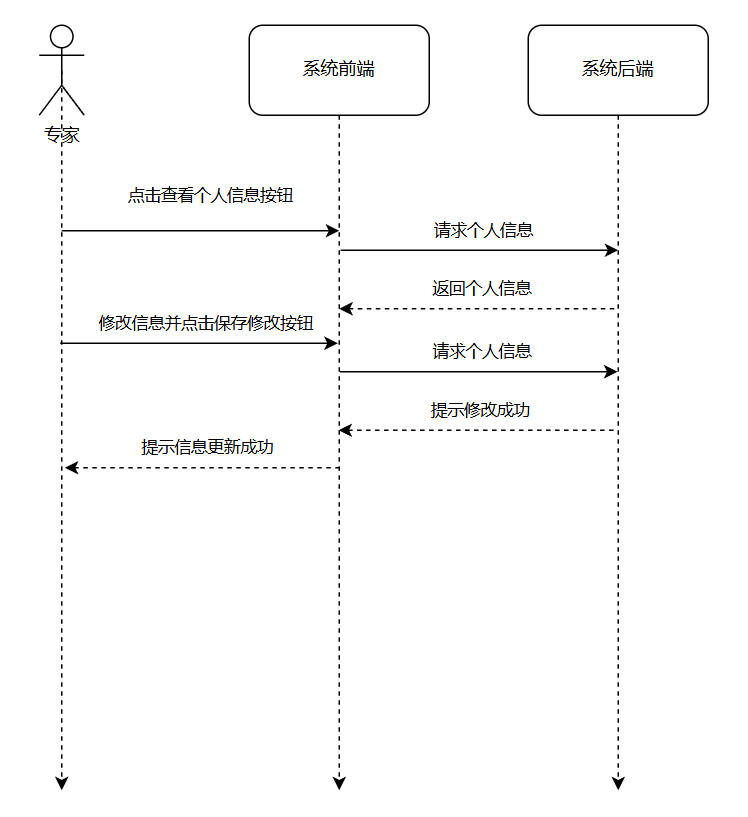
用户在前端选择用户类型，并输入身份证号、账号名、密码及确认密码，点击注册按钮后，系统前端向系统后端发送数据。后端进行数据验证，验证成功则将注册数据存入存放注册数据的数据库中，并告诉前端验证正确，前端提示用户注册成功，请等待；验证失败则告诉前端验证错误，前端提示用户错误信息。

3.2.2 登录账号



用户在前端选择用户类型，并输入账号名和密码，点击登录按钮后，系统前端向系统后端发送数据。后端进行数据验证，验证成功则告诉前端验证正确，前端提示用户登录成功；验证失败则告诉前端验证错误，前端提示用户错误信息。

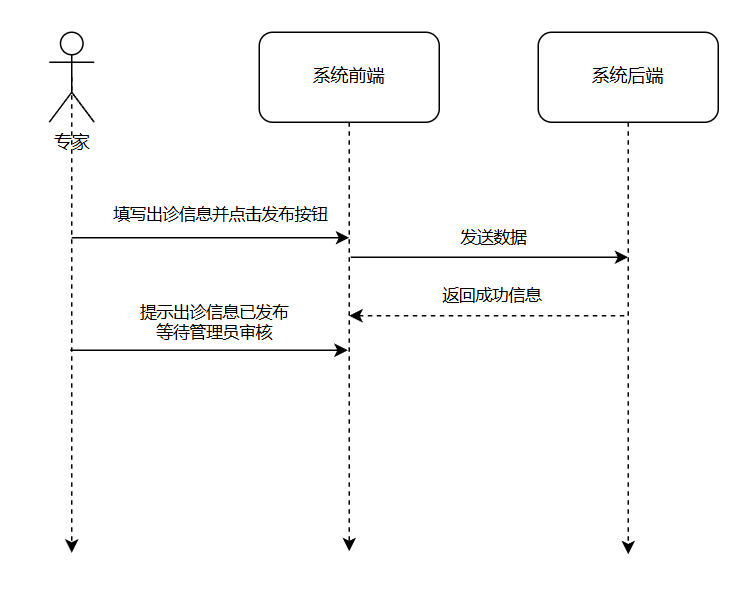
3.2.3 查看个人信息



用户点击个人信息按钮，系统前端跳转到个人信息主页，并向系统后端请求该用户的个人信息，系统后端返回该用户的个人信息并展示在系统前端中。

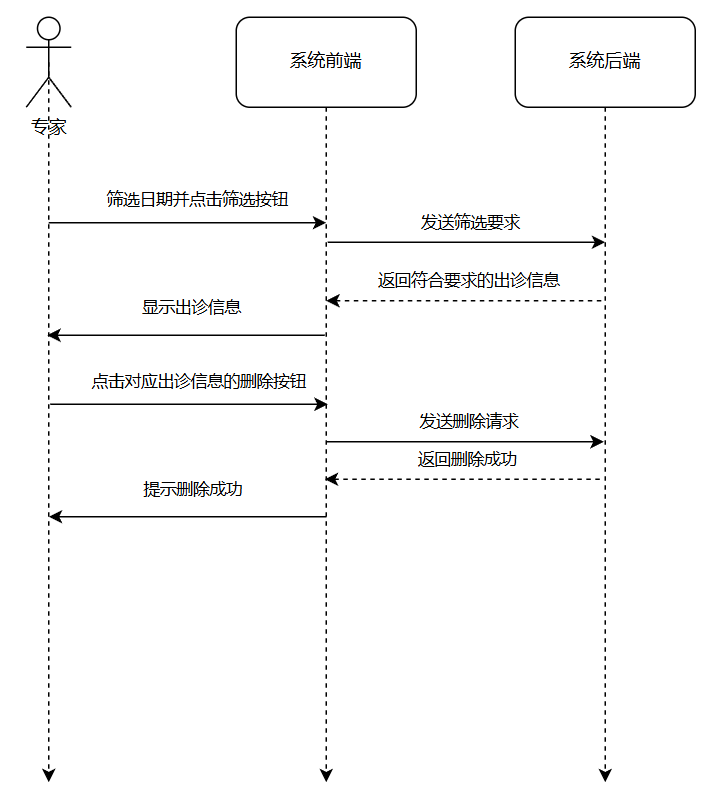
用户修改个人信息并点击保存修改按钮，系统前端将修改后的个人信息发送至后端，后端更新数据库并向前端提示修改成功，前端提示用户信息更新成功。

3.2.4 发布出诊信息



用户填写出诊信息并点击发布按钮后，系统前端向系统后端发送出诊信息的数据，系统后端将出诊信息存储在待审核出诊信息的数据库中，并向前端发送申请发布成功的信息，系统前端提示用户出诊信息已发送，请等待管理员审核。

3.2.5 查看出诊信息



用户筛选日期并点击查询按钮，系统前端向系统后端发送筛选要求，后端向前端返回符合筛选要求的出诊信息，前端向用户显示符合要求的出诊信息。

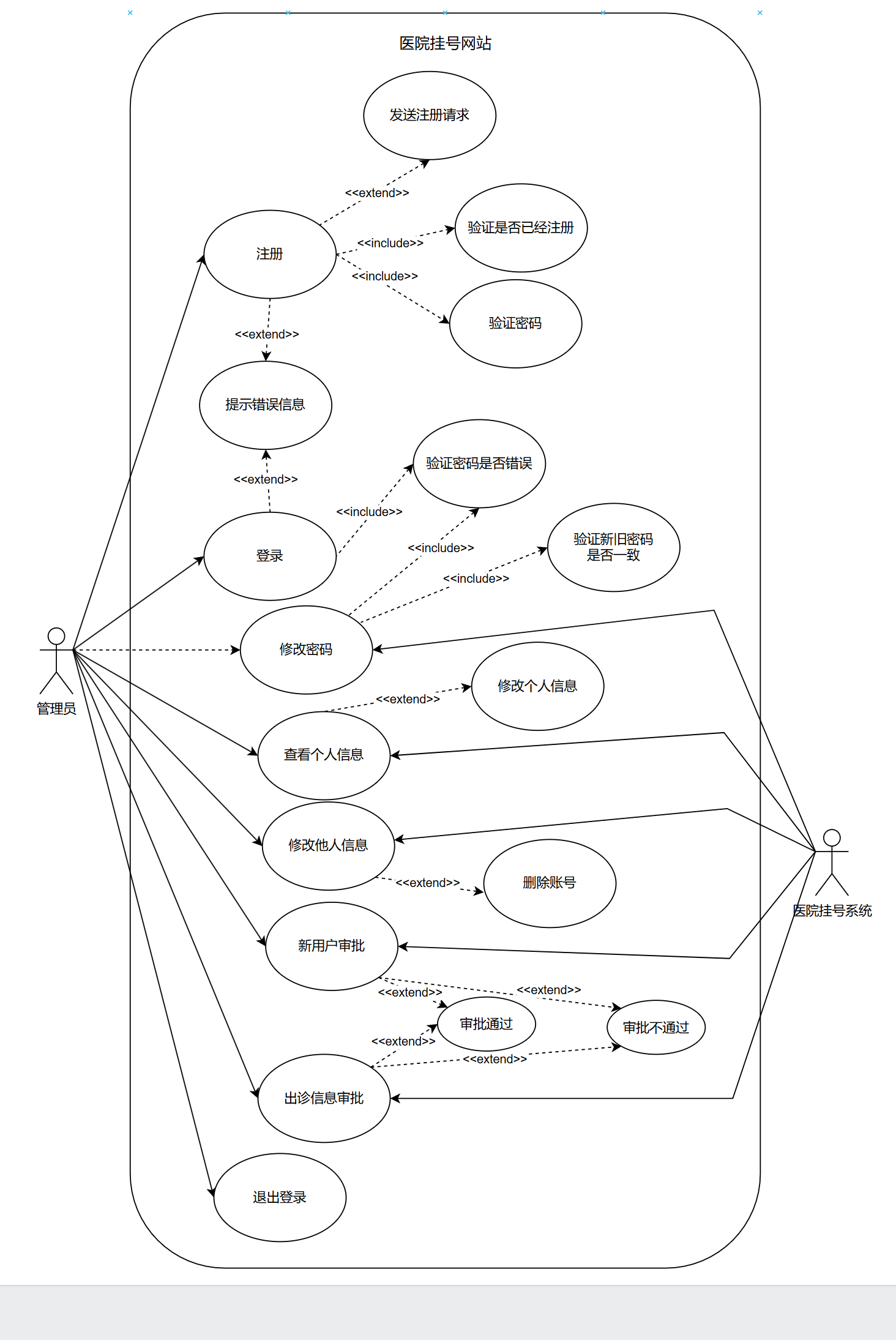
用户点击对应出诊信息的删除按钮，系统前端向系统后端发送要删除请求，包括要删除的出诊信息的具体信息，后端向前端返回删除成功信息，前端向用户提示删除成功。

3.2.6 退出登录



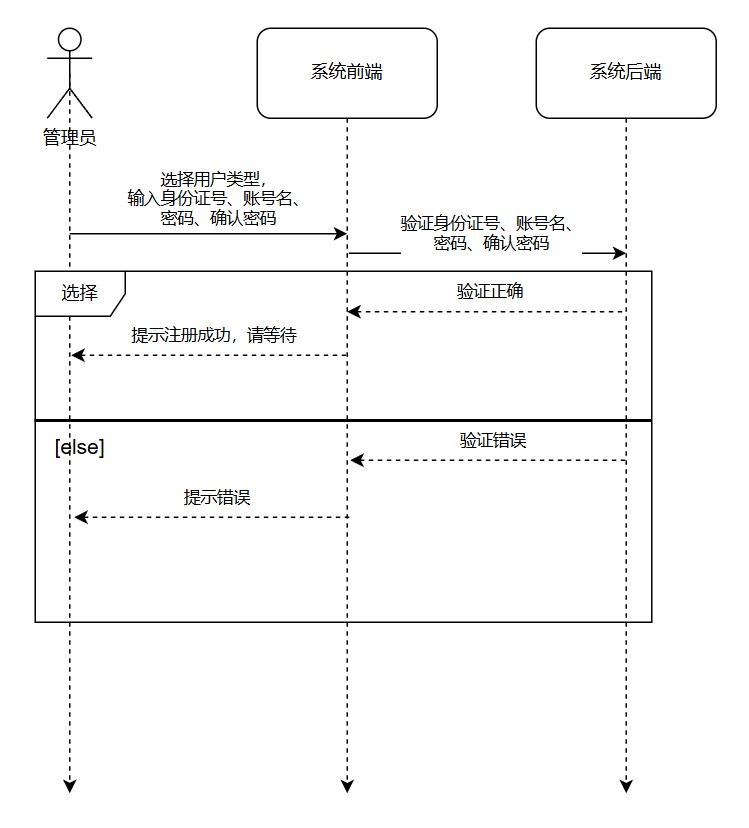
用户点击退出登录按钮后，系统前端回到系统登录首页。

3.3 管理员



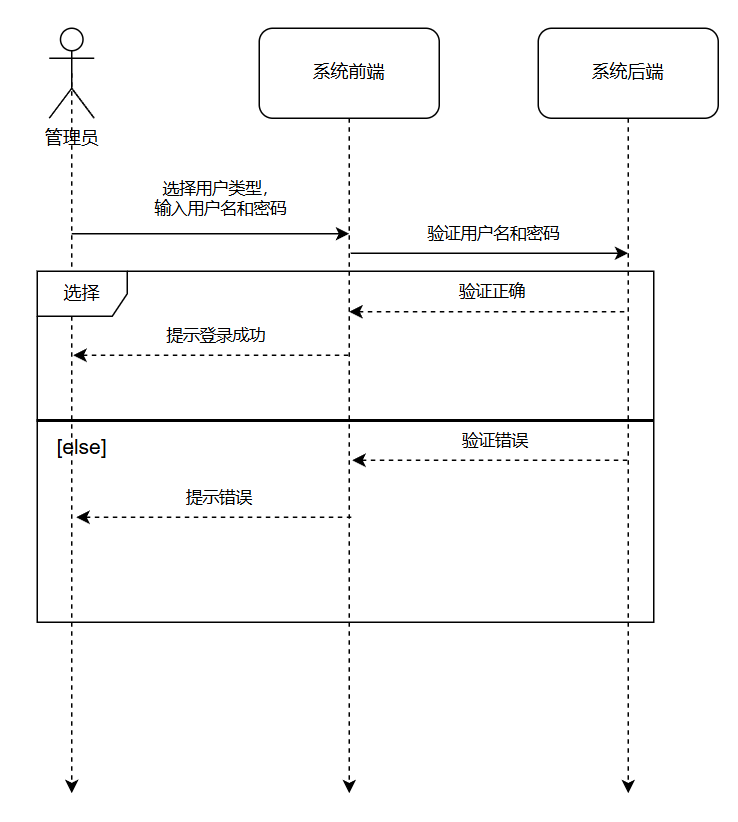
管理员可以注册、登录账号，登录账号后可以修改密码、查看个人信息、修改他人信息、审批新用户、审批出诊信息、退出登录。

3.3.1 注册账号



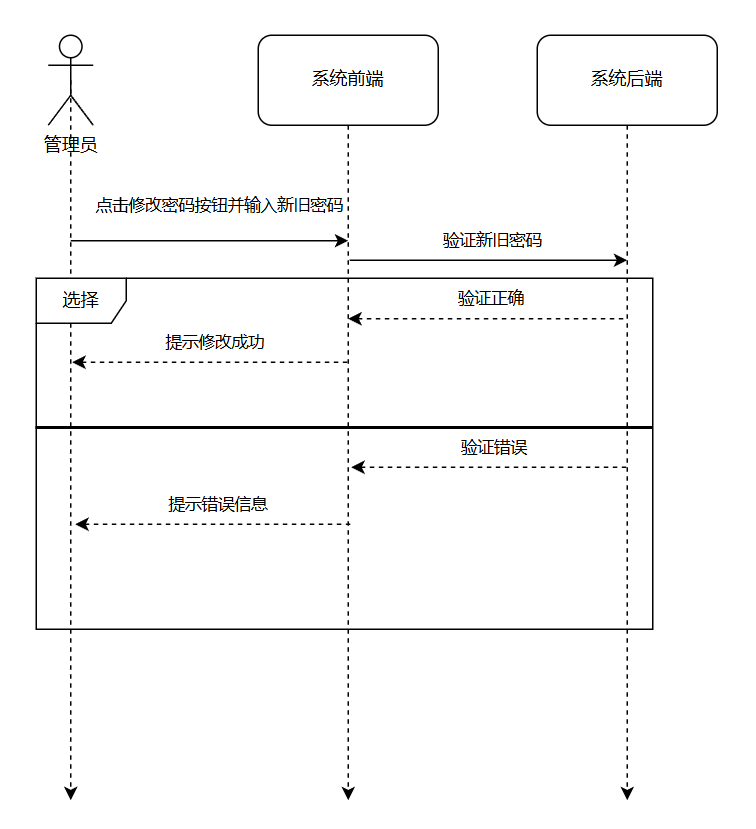
用户在前端选择用户类型，并输入身份证号、账号名、密码及确认密码，点击注册按钮后，系统前端向系统后端发送数据。后端进行数据验证，验证成功则将注册数据存入存放注册数据的数据库中，并告诉前端验证正确，前端提示用户注册成功，请等待；验证失败则告诉前端验证错误，前端提示用户错误信息。

3.3.2 登录账号



用户在前端选择用户类型，并输入账号名和密码，点击登录按钮后，系统前端向系统后端发送数据。后端进行数据验证，验证成功则告诉前端验证正确，前端提示用户登录成功；验证失败则告诉前端验证错误，前端提示用户错误信息。

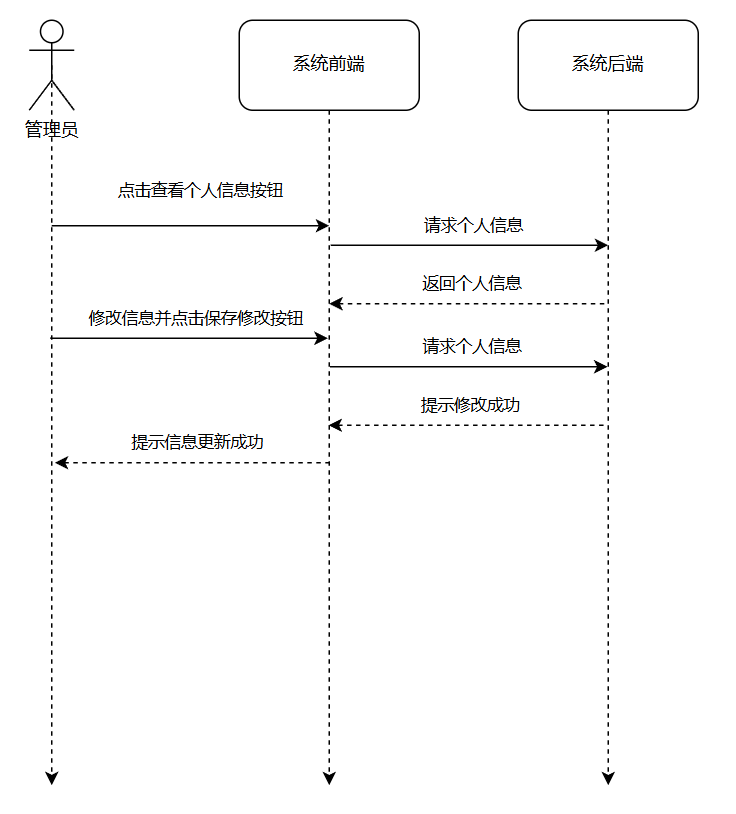
3.3.3 修改密码



用户点击修改密码后，系统前端跳转到修改密码界面。

用户输入新旧密码并点击修改密码按钮后，系统前端向系统后端发送数据。后端进行新旧密码验证，验证成功则将新密码存入数据库中，并告诉前端验证正确，前端提示用户修改成功；验证失败则告诉前端验证错误，前端提示用户错误信息。

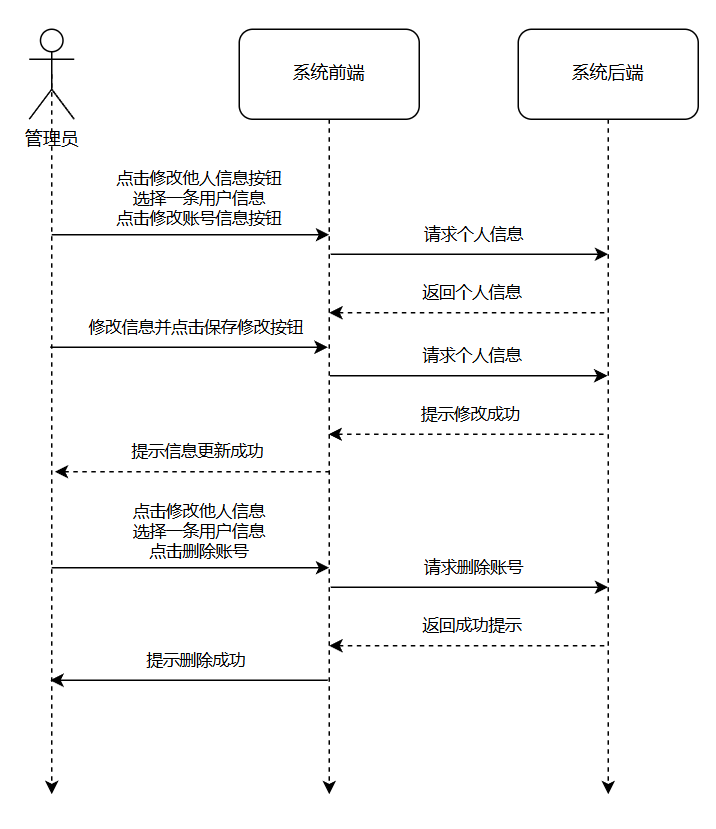
3.3.4 查看个人信息



用户点击个人信息按钮，系统前端跳转到个人信息主页，并向系统后端请求该用户的个人信息，系统后端返回该用户的个人信息并展示在系统前端中。

用户修改个人信息并点击保存修改按钮，系统前端将修改后的个人信息发送至后端，后端更新数据库并向前端提示修改成功，前端提示用户信息更新成功。

3.3.5 修改他人信息



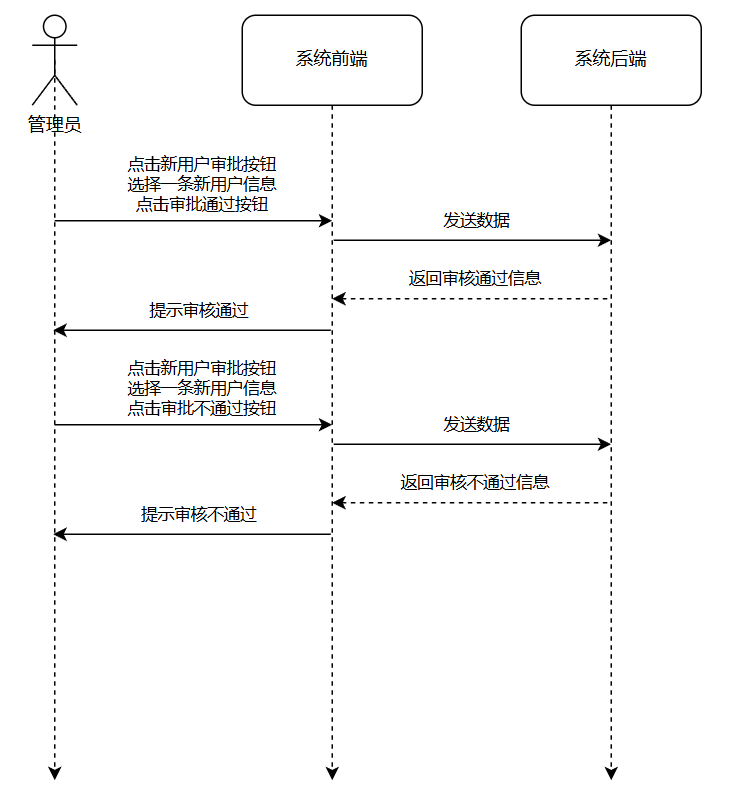
用户点击修改他人信息按钮，系统前端跳转到修改其他用户信息界面。

用户选择一条用户信息，点击修改账号信息按钮后，系统前端跳转到修改用户信息页面，并向系统后端请求被选择的用户信息，系统后端向系统前端返回被选择的用户信息。

用户修改个人信息并点击保存修改按钮，系统前端将修改后的个人信息发送至后端，后端更新数据库并向前端提示修改成功，前端提示用户信息更新成功。

用户选择一条用户信息，点击删除账号按钮后，系统前端向系统后端发送被选择的用户信息并请求删除，系统后端数据库删除该用户并向系统前端返回成功提示，系统前端向用户提示删除成功。

3.3.6 新用户审批

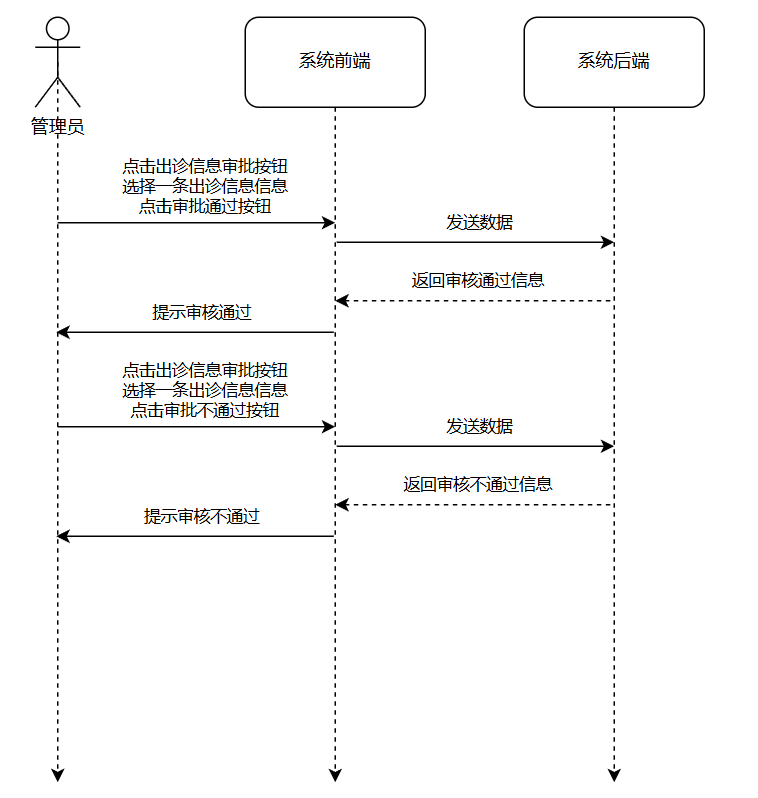


用户点击新用户审批按钮，系统前端跳转到审批新用户界面。

用户选择一条新用户信息并点击审批通过按钮，系统前端向系统后端发送被选择的新用户的具体信息和审核通过请求，系统后端将审批通过的新用户加入到用户对应类型的数据库中，并将该新用户从存放注册数据的数据库中删除，并向前端返回审核通过信息，系统前端向用户显示审核通过。

用户选择一条新用户信息并点击审批不通过按钮，系统前端向系统后端发送被选择的新用户的具体信息和审核不通过请求，系统后端将该新用户从存放注册数据的数据库中删除，并向前端返回审核不通过信息，系统前端向用户显示审核不通过。

3.3.7 出诊信息审批

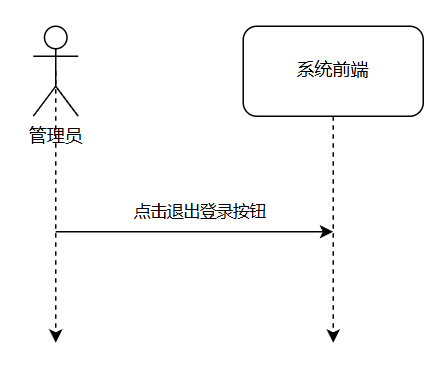


用户点击出诊信息审批按钮，系统前端跳转到审批出诊信息界面。

用户选择一条出诊信息信息并点击审批通过按钮，系统前端向系统后端发送被选择的出诊信息的具体信息和审核通过请求，系统后端将审批通过的出诊信息加入到存放出诊信息的数据库中，并将该出诊信息从待审核出诊信息的数据库中删除，并向前端返回审核通过信息，系统前端向用户显示审核通过。

用户选择一条出诊信息信息并点击审批不通过按钮，系统前端向系统后端发送被选择的出诊信息的具体信息和审核通过请求，系统后端将该出诊信息从待审核出诊信息的数据库中删除，并向前端返回审核通过信息，系统前端向用户显示审核通过。

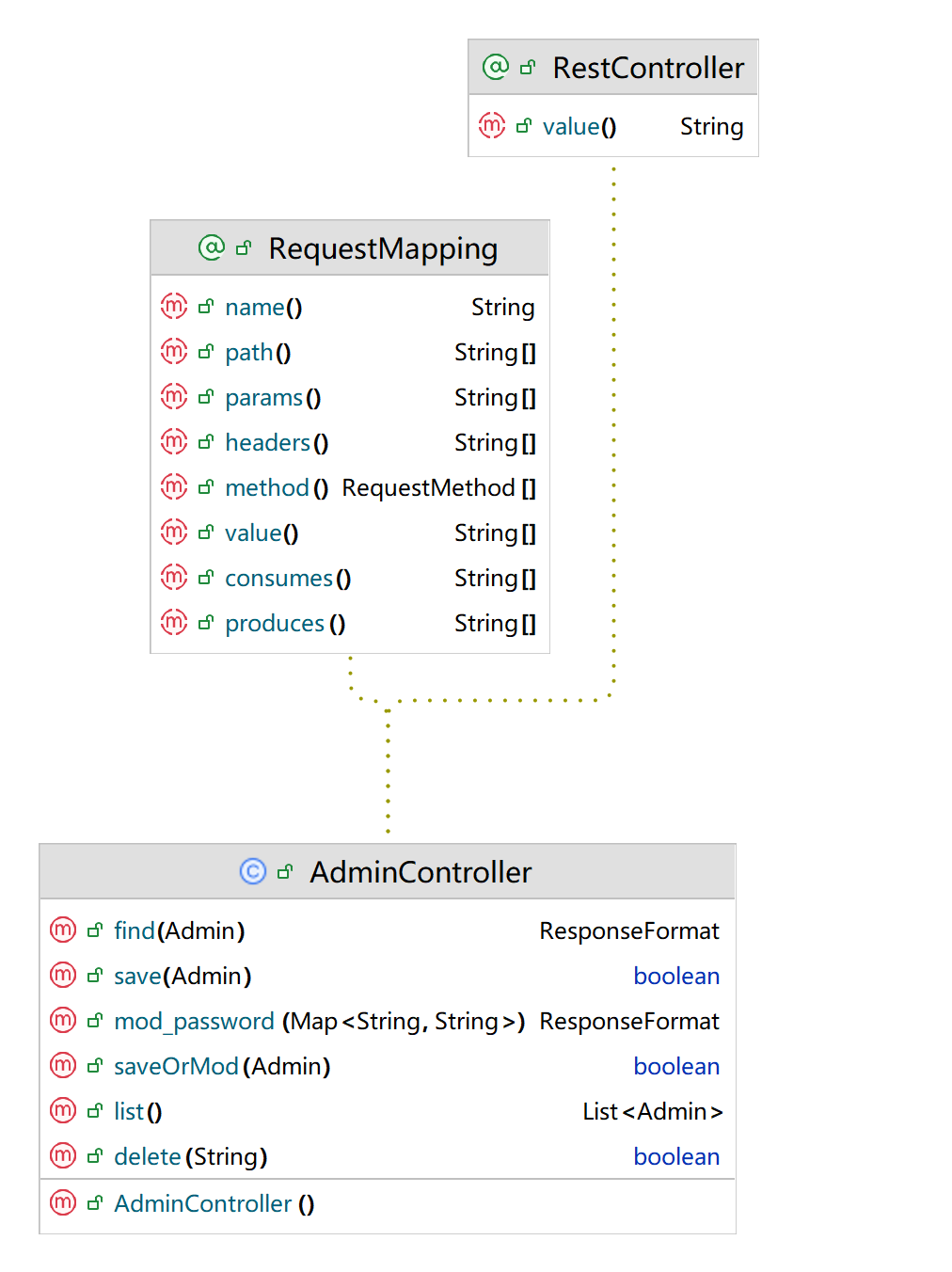
3.3.8 退出登录



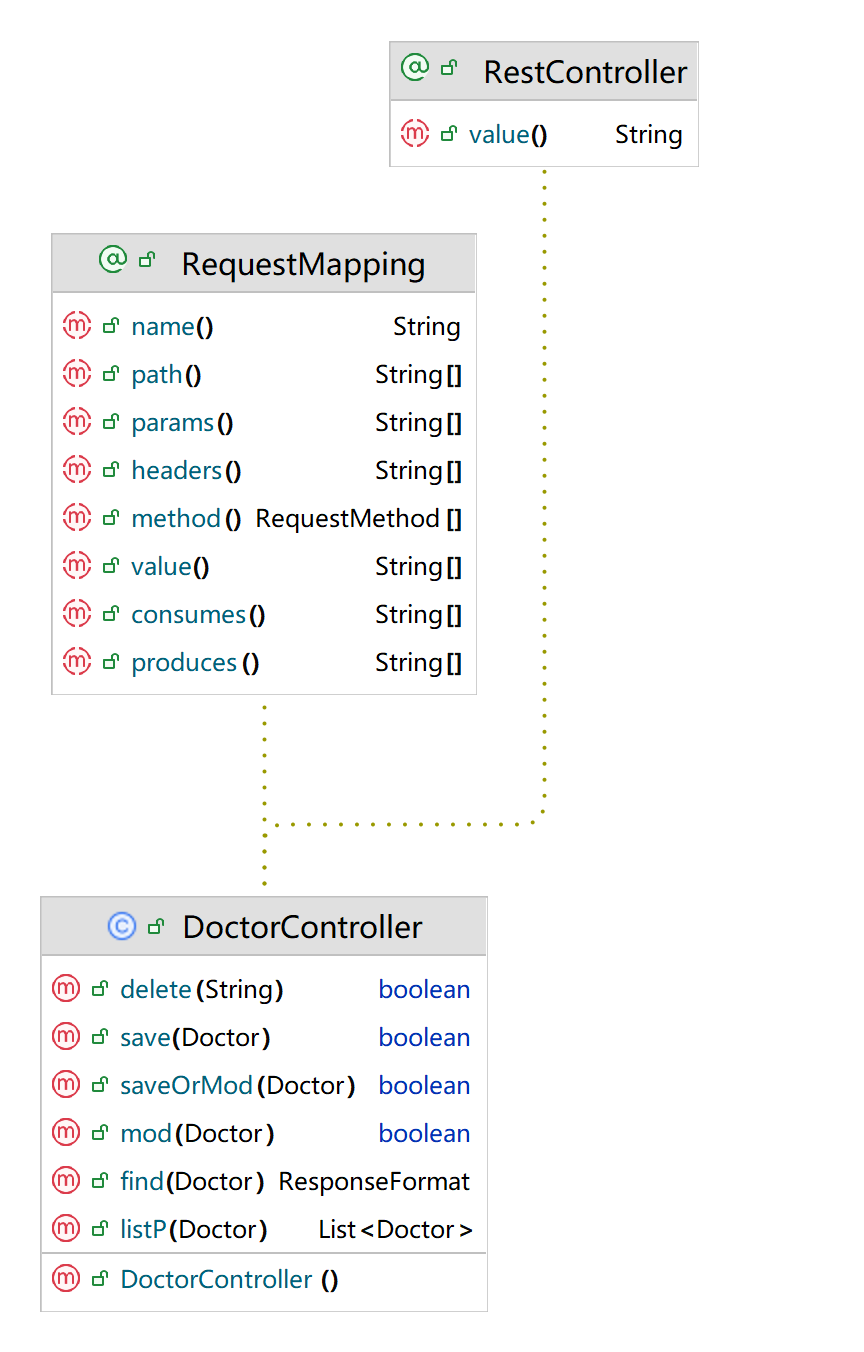
用户点击退出登录按钮后，系统前端回到系统登录首页。

4 面向对象分析

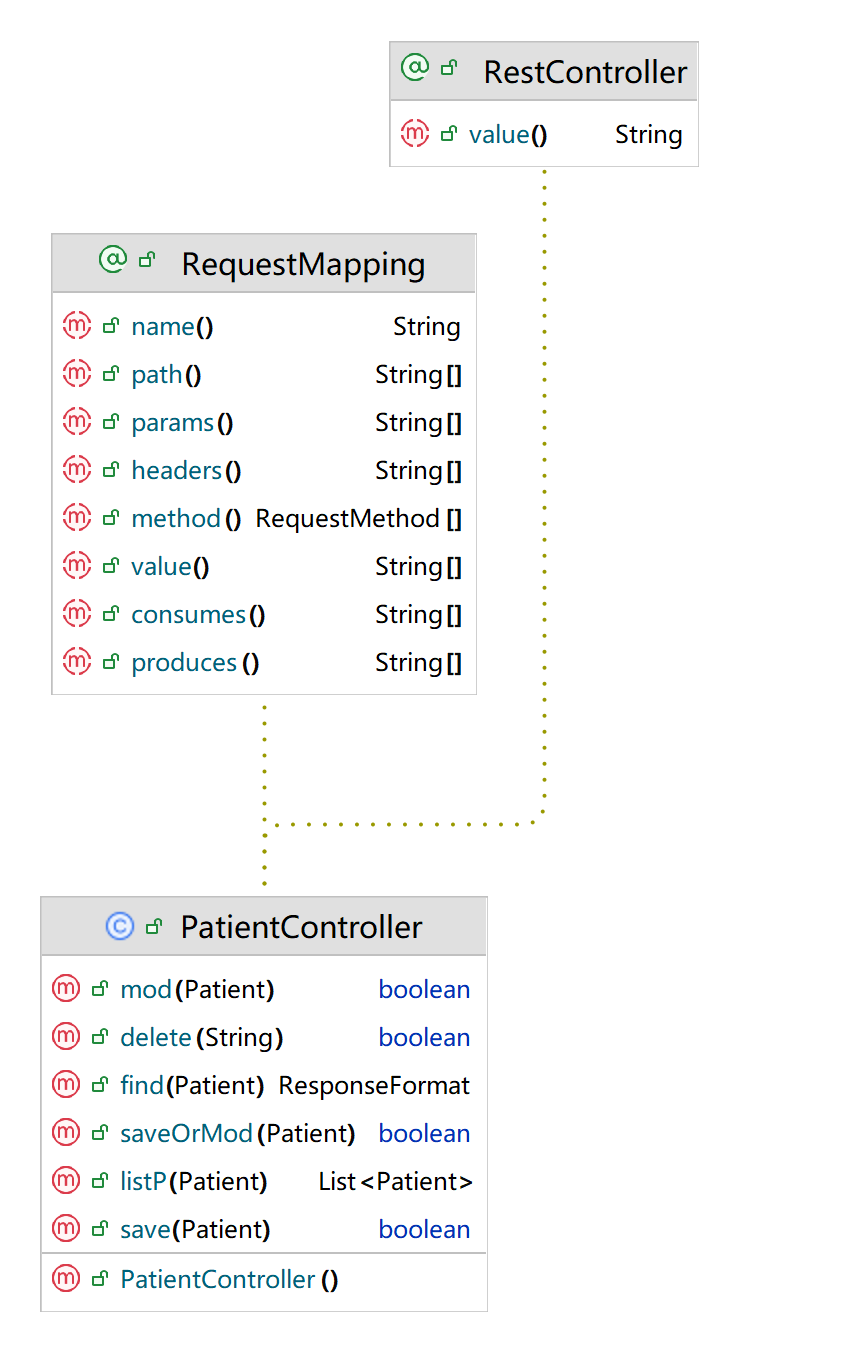
4.1 Admin类



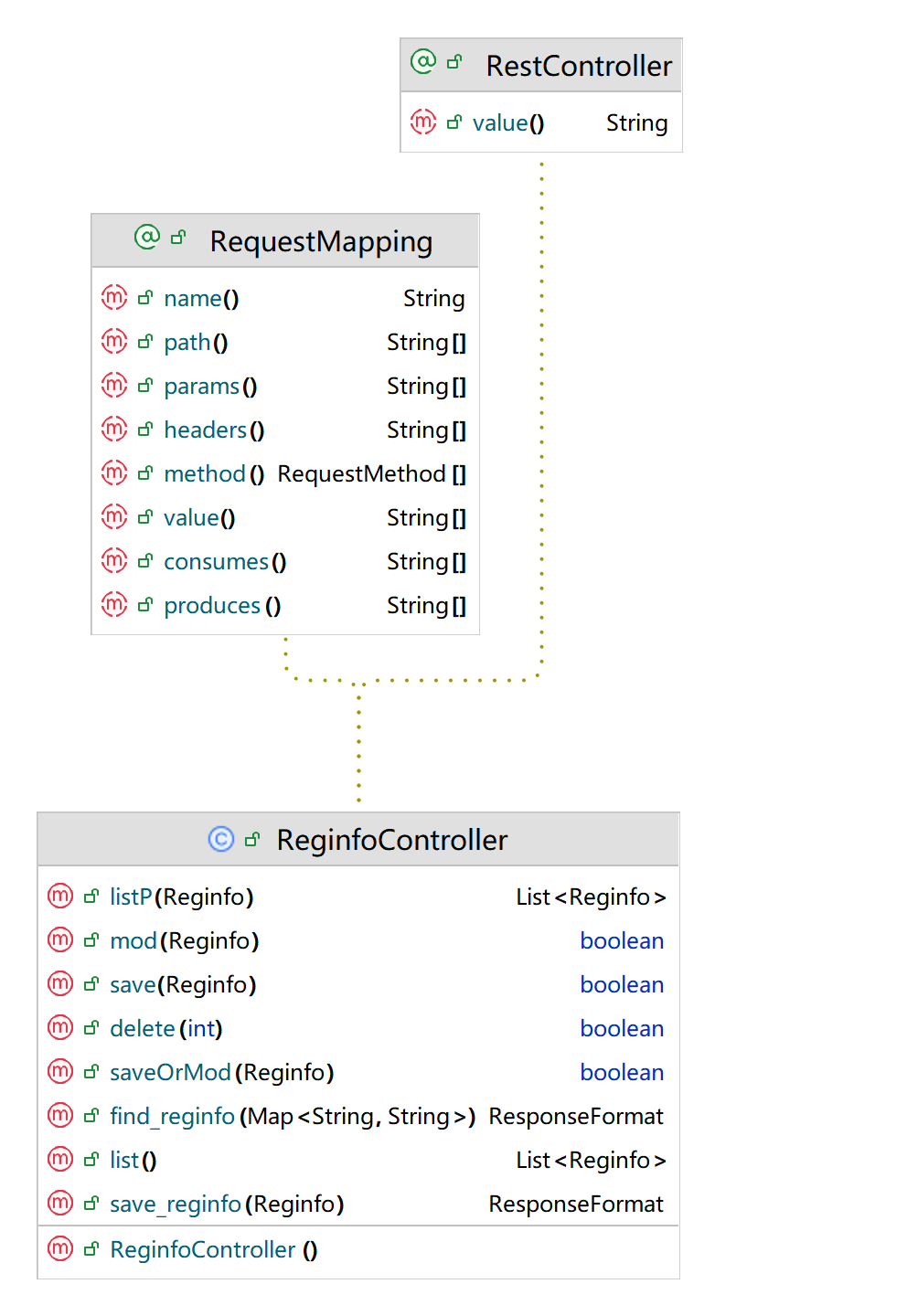
4.2 Doctor类



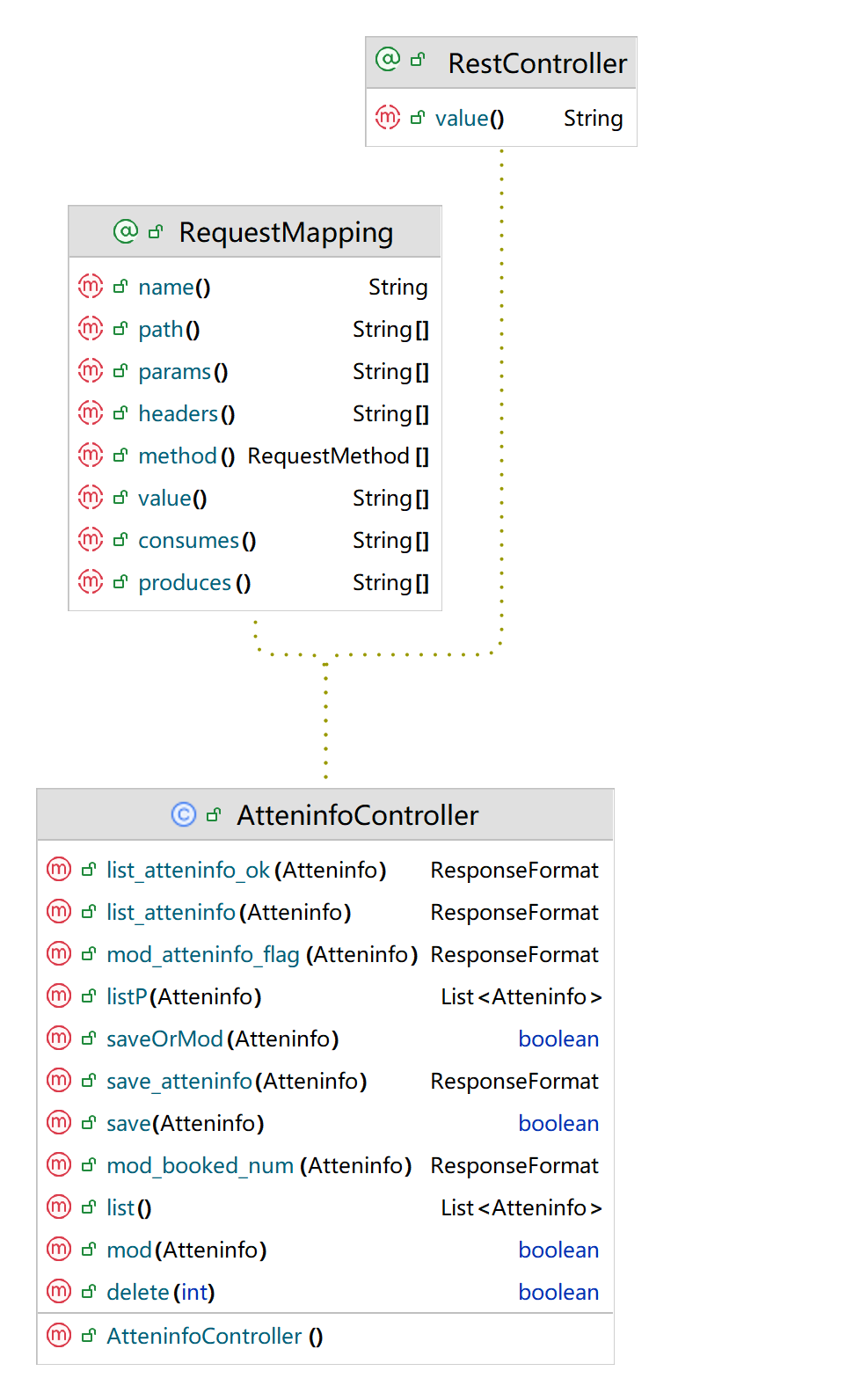
4.3 Patient类



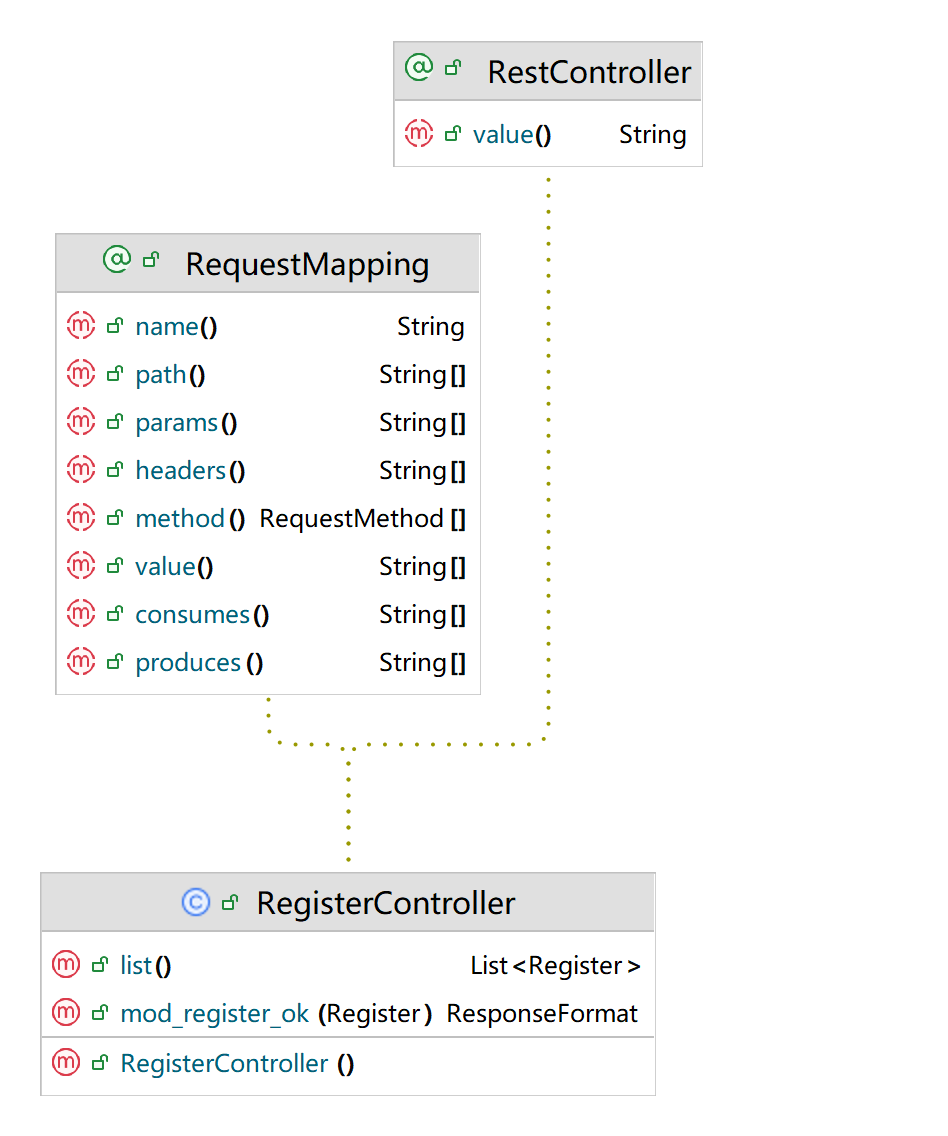
4.4 Reginfo类



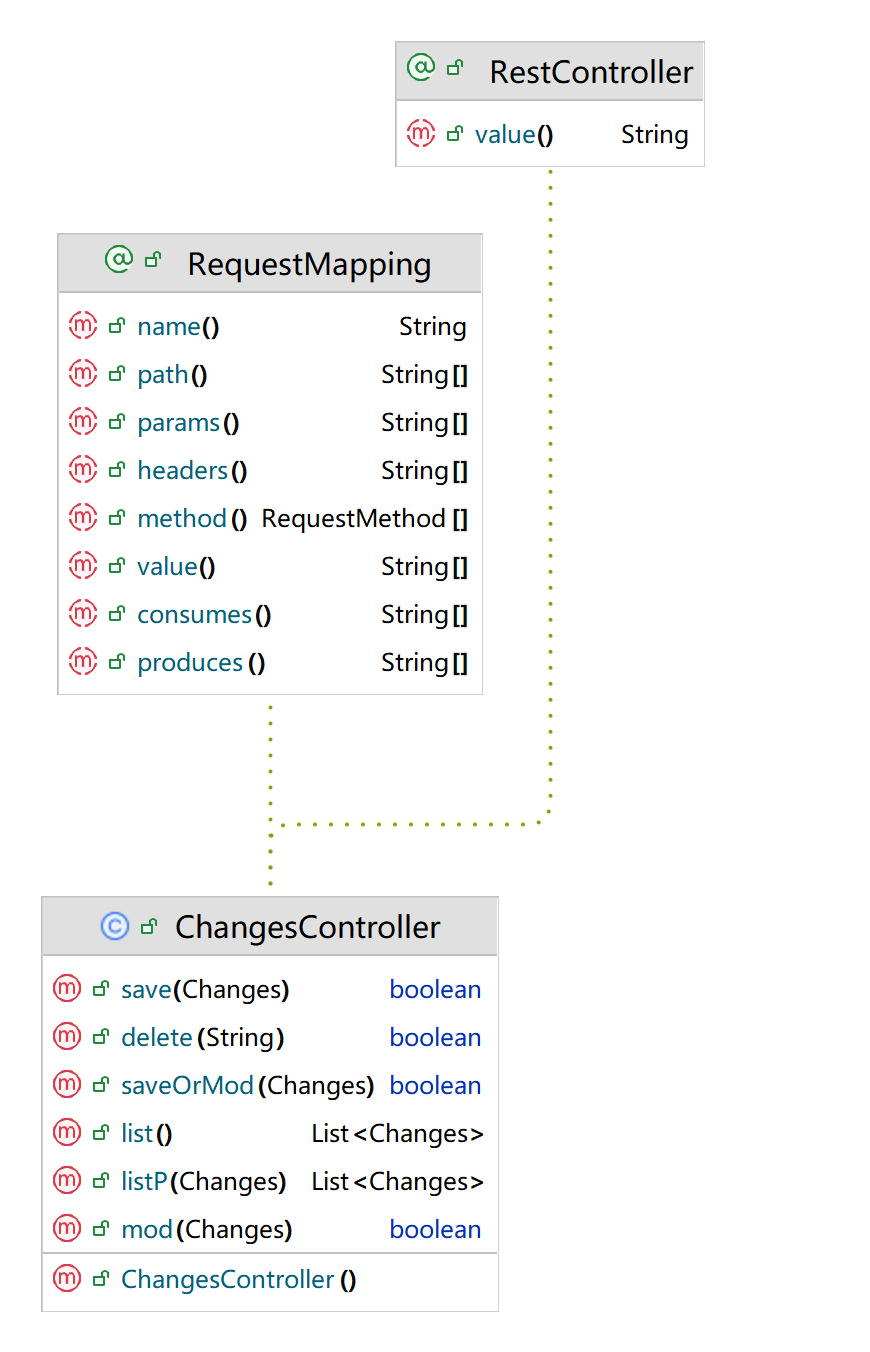
4.5 Atteninfo类



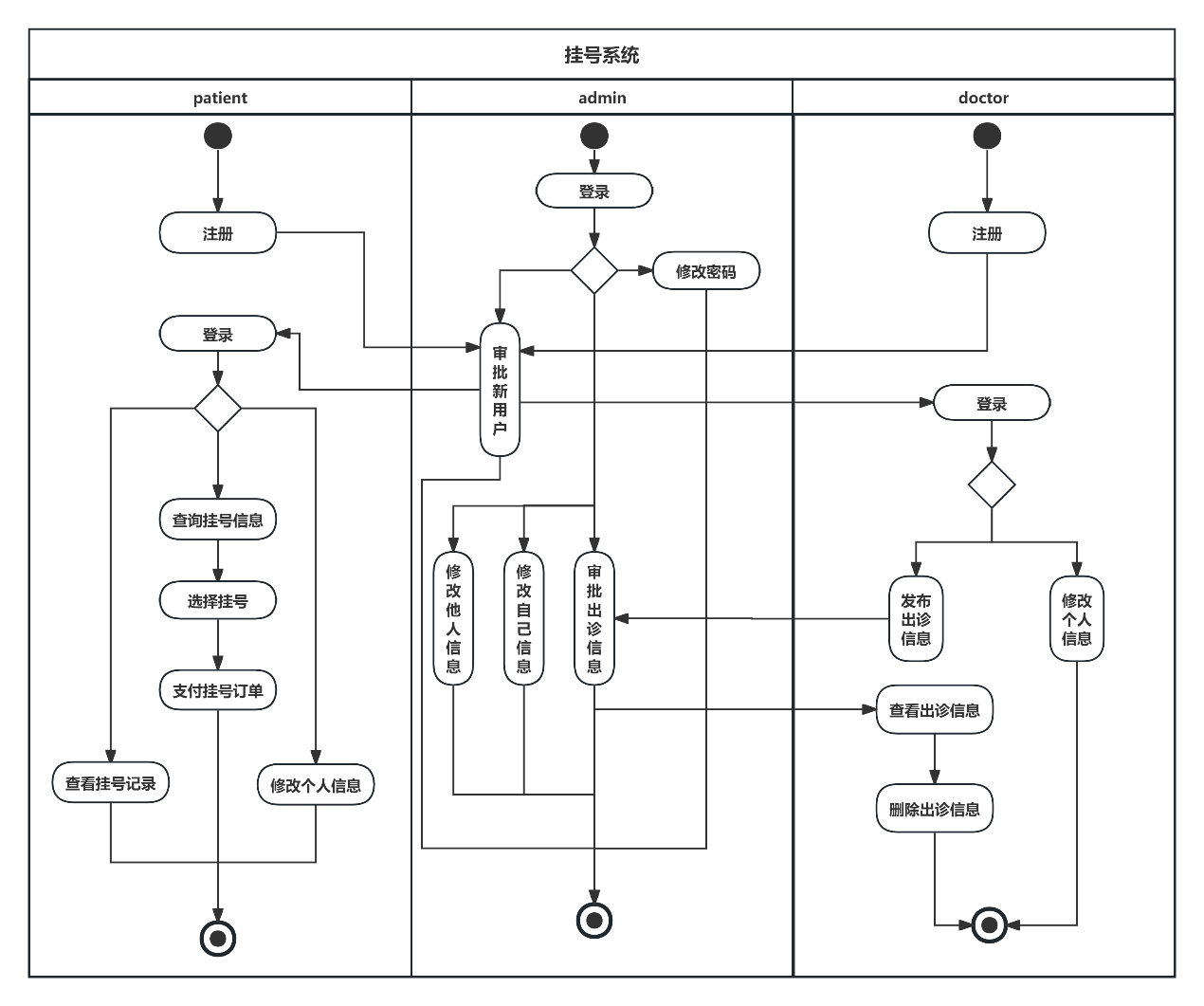
4.6 Register类



4.7 Changes类



5 面向对象设计



系统活动图如上图所示。顺序图痛需求分析中的顺序图，类图同面向对象分析的类图。

普通用户（患者，patient）注册后须待管理员审核通过。审核通过后可进行登录。登录后可查看挂号记录、查询挂号信息、修改个人信息。在查询挂号信息后，普通用户可以选择挂号，然后支付挂号订单。

管理员（admin）注册后须待管理员审核通过（图中省略）。管理员登录后可以修改密码、修改自己信息、修改他人信息、审批新用户、审批出诊信息。

专家（医生，doctor）注册后须待管理员审核通过。审核通过后可进行登录。登录后可发布出诊信息、修改个人信息。发布出诊信息后，须待管理员审核通过，管理员审核通过后可查看已通过审核的出诊信息，然后可以选择删除出诊信息。

6 面向对象测试

6.1 系统首页



运行程序，来到医院挂号系统首页。

点击注册按钮可以前往注册页面，点击登录按钮可以前往登录页面。

6.1.1 注册帐号



在注册页面，用户需要选择注册的账号类型，然后输入身份证号、账号名、密码，并再次输入密码。已有该类型的账号名或两次输入的密码不同将无法注册。若注册成功则会提示用户注册成功，请等待管理员审核。

6.1.2 登录账号



在登录页面，用户需要选择登录的账号类型，然后输入用户名和密码。该类型的用户名不存在或密码错误将无法登录。登录成功后将提示登录成功，并跳转到对应账号类型的系统。若选择账号类型为普通用户，则跳转到患者系统；若选择账号类型为医生，则跳转到专家系统；若选择账号类型为管理员，则跳转到管理员系统。

6.2 患者系统



在患者系统首页，用户可以挂号、查看个人信息、查看挂号记录、退出登录。

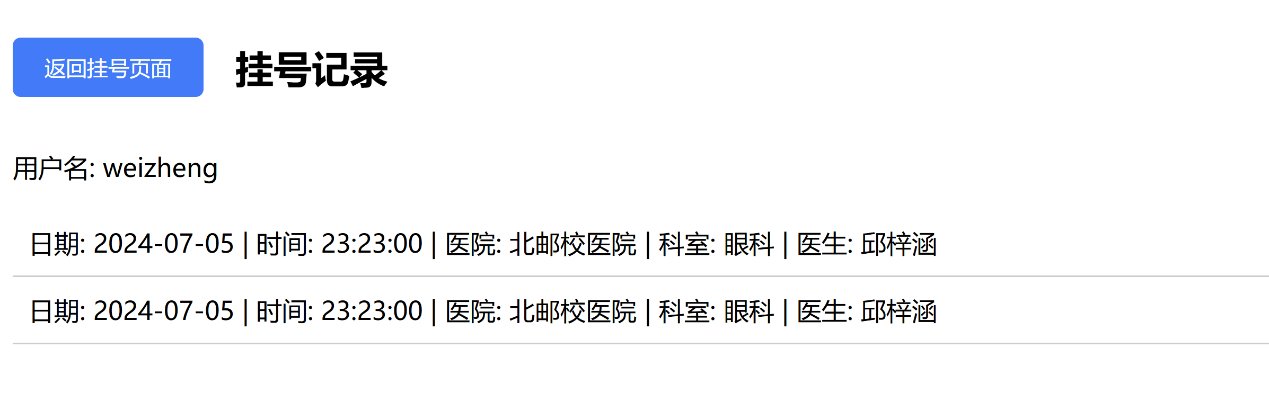
6.2.1 挂号功能



用户需要选择相应挂号的日期、医院、科室，然后选择医生并点击查询按钮，系统将为用户展示所有符合要求的挂号信息。

可以点击选择挂号，并点击确认支付，以完成挂号。

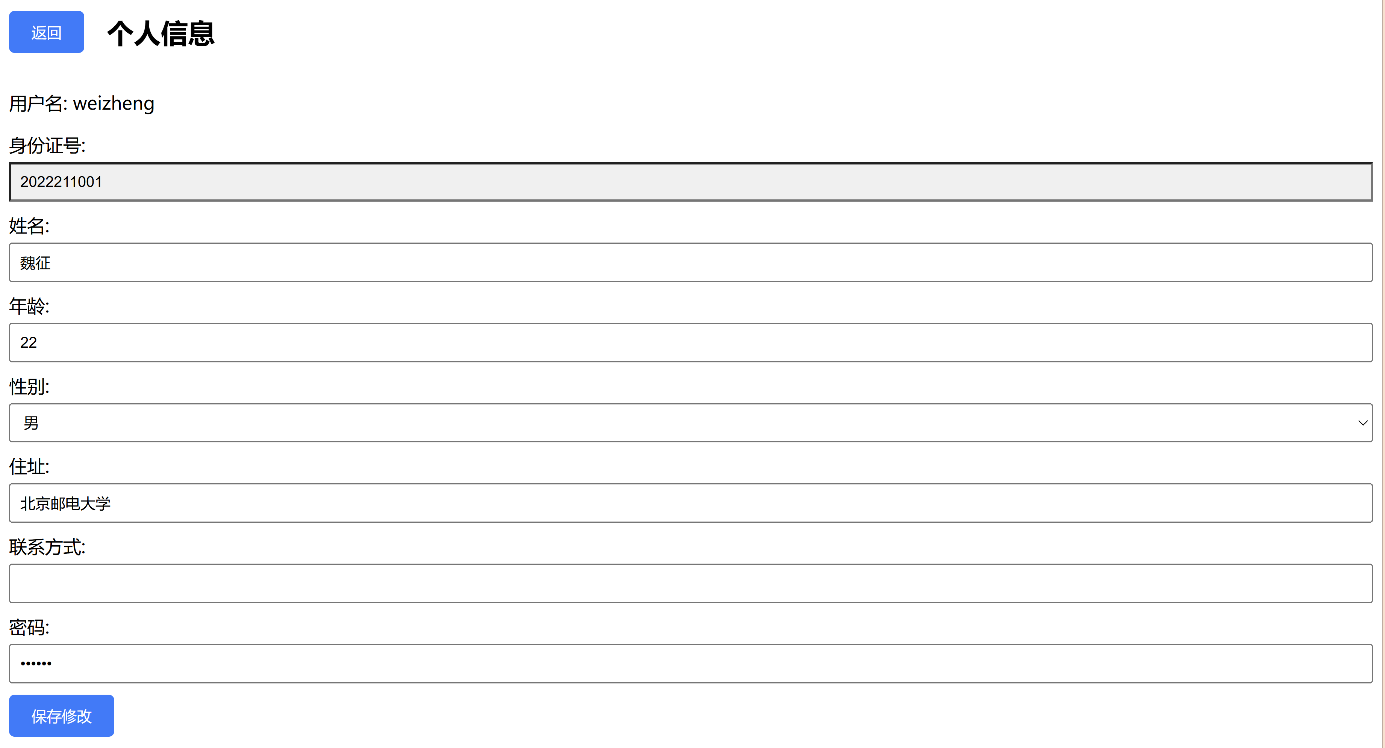
6.2.2 查看挂号记录



用户在患者首页点击挂号记录按钮，即可查看自己的挂号记录。

点击返回按钮可以返回患者首页。

6.2.3 修改个人信息



用户在患者首页点击个人信息按钮，即可查看并修改自己的个人信息。

进入个人信息页面，用户可以修改自己的个人信息，并点击保存修改按钮。系统提示更新成功后即修改个人信息成功。

点击返回按钮可以返回患者首页。

6.2.4 退出登录

用户点击退出登录按钮，即可退出登录，页面将回到系统首页。

6.3 专家系统



在专家系统首页，用户可以发布出诊信息、查看出诊信息、查看个人信息、退出登录。

6.3.1 发布出诊信息

用户选择想要发布出诊信息的出诊日期、出诊时间和总挂号数量，然后点击发布按钮，待管理员审核之后即可成功发布。

6.3.2 查看出诊信息

用户选择想要查询出诊信息的出诊日期，点击查询按钮即可查询到符合要求的出诊信息。

点击删除按钮即可删除该出诊信息。

6.3.3 查看个人信息

用户在专家首页点击个人信息按钮，即可查看并修改自己的个人信息。

进入个人信息页面，用户可以修改自己的个人信息，并点击保存修改按钮。系统提示更新成功后即修改个人信息成功。

点击返回按钮可以返回专家首页。

6.3.4 退出登录

用户点击退出登录按钮，即可退出登录，页面将回到系统首页。

6.4 管理员系统



在管理员系统首页，用户可以修改密码、修改自身信息、修改他人信息、审批新用户、审批出诊信息、退出登录。

6.4.1 修改密码

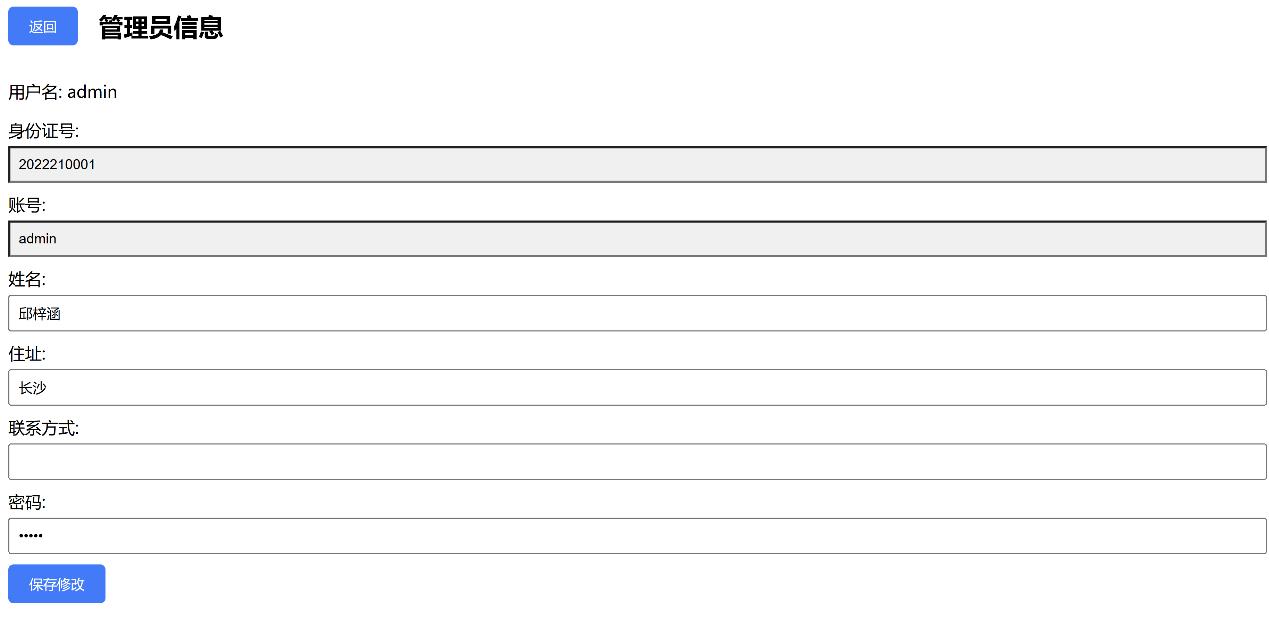


点击修改密码按钮，进入修改密码页面。

用户需要输入旧密码和新密码。旧密码错误或者新密码与旧密码一致将导致修改密码失败。点击修改密码，提升修改成功后即成功修改密码。

点击返回首页按钮可以返回管理员首页。

6.4.2 修改自身信息



点击修改自身信息按钮，即可查看并修改自己的个人信息。

进入个人信息页面，用户可以修改自己的个人信息，并点击保存修改按钮。系统提示更新成功后即修改个人信息成功。

点击返回按钮可以返回管理员首页。

6.4.3 修改他人信息



点击修改他人信息按钮，即可修改或删除他人信息。

用户选择其中一个账号，可以修改该账号信息或删除该账号。

点击修改账号信息按钮，即可跳转到修改账号信息页面，用户可以修改该账号的个人信息，并点击保存修改按钮。系统提示更新成功后即修改个人信息成功。

点击删除账号信息按钮，即可删除该账号。

6.4.4 审批新用户



点击审批新用户按钮，即可跳转到审批新用户界面。

用户选择其中一个新用户，可以选择审批通过或审批不通过。

6.4.5 审批出诊信息



点击审批出诊信息按钮，即可跳转到审批出诊信息界面。

用户选择其中一个出诊信息，可以选择审批通过或审批不通过。

7 需求新增

7.1 患者违约信息

医院提出需要增加统计患者的违约信息。180天内3次挂号不去看病的患者将被取消未来360天的预约挂号功能，并根据违约情况等建立患者信用分。

对于取消预约挂号功能，首先需要判定患者是否有去看病。解决方案是在患者系统的挂号记录页面，对于每条挂号记录新增一个去看病按钮，仅当该挂号记录的日期为当前系统日期时可用。点击该按钮即代表患者有去看病。

因此，判断用户是否失去未来360天的预约挂号资格的方法是查看过去180天内该用户是否拥有三条或以上的未去看病的记录。有则失去未来360天的预约挂号资格，否则对预约挂号功能无影响。

信用分将成为患者属性的一部分，在患者系统查看个人信息时展示且无法修改。初始时为80分，每条未去看病的挂号记录扣5分，每条未去看病的挂号记录且该号源来自拍卖系统的扣10分，每条去看病的挂号记录加1分，上限为100分，下限为0分。

7.2 号源拍卖功能

医院推出部分号源竞价拍卖功能，标准是价格和患者信用分乘积最大者得。

分析需求，首先要能推出用于竞价拍卖的号源。在专家系统中的发布出诊信息功能中，新增需要选择出诊信息为预约型还是竞价拍卖型，若为预约型则出现在患者系统的预约挂号功能中，若为竞价拍卖型则出现在患者系统的竞价拍卖功能中。除此之外，若为竞价拍卖型，还需设置竞价拍卖结束时间，该事件需要在该出诊信息的日期之前。

相应的，患者系统需要新增竞价拍卖功能。刚进入该页面时，系统将为用户展示所有的竞价拍卖型号源，用户可以通过选择日期、医院、科室、医生筛选号源。选择相应号源便跳转到该号源的竞价拍卖界面。

竞价拍卖界面首先显示该号源的基本信息，包括该挂号信息的日期、所属医院、对应科室、对应医生。然后显示该号源的竞价拍卖结束日期、目前最高价（价格\*信用分\*0.01）。用户可以输入自己的出价，系统将同步计算实际出价（价格\*信用分\*0.01）。若出价低于或等于目前最高价则提示竞价失败，若出价高于目前最高价则提示竞价成功，并更新该竞价拍卖号源的目前最高价。该页面也应该有返回按钮，用于返回首页。

7.3 用户信息新增

一方面，其他医院也希望加入该系统，所以专家需要增加所属医院信息，挂号时首先选择医院。另一方面，管理员分为两级，某医院管理员和系统管理员。

对于增加所属医院信息并在挂号时首先选择医院，本系统已实现。

对于管理员分为某医院管理员和系统管理员。在登录系统的登录功能及注册功能中，目前账号类型为管理员的属于系统管理员，系统管理员的注册、登录、登录后的系统管理员系统与前文管理员的注册、登录、登陆后的管理员系统相同。新增一种账号类型为医院管理员。

在注册功能中，当用户选择注册为医院管理员时，除了要输入身份证号、用户名、密码、确认密码以外，还需要输入管理的医院，这一项在系统管理员审批时也会展示。

在登录功能中，当用户选择登录为医院管理员时，除了要输入用户名和密码以外，还需要输入管理的医院，用于登录确认。所属医院无该用户名的医院管理员时同样无法登录。

登录后进入医院管理员系统。实际上，医院管理员系统为系统管理员系统的阉割版，拥有修改密码、修改自身信息、修改他人信息、退出登录的功能。与系统管理员系统不同的是，医院管理员系统没有新用户审批功能，且在修改他人信息中，仅能修改该医院管理员所属医院的医生的账号信息。

需要注意的是，在修改自身信息功能中，用户所属的医院将被展示，但无法修改。

7.4 相似挂号推荐

用户点击号挂完了的医生，可以推荐其他就诊时间或者其他相关医生信息。

当用户点击号挂完了的号源时，将修改医生筛选条件为所有医生，修改日期并再次查询。每次修改日期时检测该日期下是否有剩余号源，若有则不再修改并给出该日期下的查询结果。若没有则继续修改日期，直到达到预约的三天时限，此时将提示无相似号源。

8 项目总结

8.1 总结体会

在这个项目中，我们经历了从需求分析到系统设计再到功能实现的完整开发过程。这不仅提高了我们的编程技能，也深化了我们对软件工程理论和实践的理解。

通过详细的需求分析，我们明确了系统各个模块的功能和交互方式，为后续的设计和实现奠定了坚实的基础。

在项目开发过程中，团队成员各司其职，密切合作，共同解决了许多技术难题。我们体会到良好的沟通和有效的协作是项目成功的关键。

通过选择Spring Boot、Vue.js和MySQL等开发环境或编程语言，我们快速搭建了系统的前后端架构。这些技术不仅提高了开发效率，也保证了系统的稳定性和扩展性。

在开发过程中，我们遇到了许多技术难题，如并发访问处理、数据一致性维护等。通过不断地调试和优化，我们逐步解决了这些问题，系统性能和可靠性得到了显著提升。

8.2 后续改进

尽管我们已经实现了系统的基本功能，但仍有许多可以改进和扩展的地方。

8.2.1 性能优化

针对每天早上8:00访问量集中的情况，我们可以进一步优化系统性能，如引入缓存机制、优化数据库查询等，以提高系统响应速度和处理能力。

8.2.2 安全性增强

加强系统的安全性措施，如增加数据加密、强化身份验证等，保护用户的隐私和数据安全。

8.2.3 身份信息的唯一性

现在的身份信息等用户信息，都为用户自己设定的，虽然也可以由管理员进行修改，但在实际应用上，会造成需多虚假信息。对于这个问题的改进，我们提出采用实名认证的办法解决，即：每次登录之后，首先进行实名认证，输入身份证和对应的名字、住址、性别，保证这些基本信息的正确性。

8.2.4 系统的用户友好性

目前的系统还很简陋，没有相关医院的信息展示，用户也无法查看医生的个人主页等等问题。出于对用户体验的考虑，后续我们提出美化前端界面、建立医院主页、医生主页等用户需求页面。

9 分工情况

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 数据库  搭建 | 后端功能开发 | 前端界面开发 | 前后端  交互连接 | 报告撰写 |
| 廖轩毅 | 22% |  | 30% |  | 70% |
| 张晨阳 |  |  | 70% | 50% | 10% |
| 魏征 | 26% | 33% |  | 25% | 10% |
| 邱梓涵 | 26% | 33% |  | 25% | 10% |
| 李梓健 | 26% | 33% |  |  |  |