

中国大学生计算机设计大赛



软件开发类作品文档简要要求

作品编号： 2025071223

作品名称： “夕阳”养老院管理应用

作 者： 张利飞、何洁、赵晓英、王文豪

版本编号： V0.0.1

填写日期： 2025.5.9

填写说明：

- 1、本文档适用于所有涉及软件应用与开发类的各个小类作品，包括：(1) Web 应用与开发 (2) 管理信息系统 (3) 移动应用开发（非游戏类）(4) 算法设计与应用 (5) 企业赛项；
- 2、本文档为简要文档，不宜长篇大论，需简明扼要，建议设计二级目录，逻辑性强；
- 3、一级标题采用二号黑体，居中，二级标题采用三号黑体，靠左，根据需要可以设计三级标题，正文一律用五号宋体；
- 4、提交文档时，以 PDF 格式提交本文档；
- 5、本文档内容是正式参赛内容组成部分，务必真实填写。如不属实，将导致奖项等级降低甚至终止本作品参加比赛。
- 6、项目中涉及应用人工智能大模型的内容，务必在各个部分说明清楚。

目 录

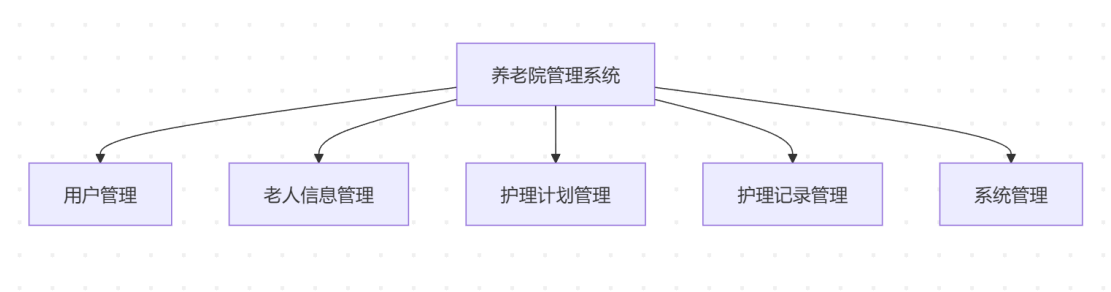
第一章 需求分析	3
第二章 概要设计	3
2.1 系统功能模块划分	3
2.2 模块层次结构	3
2.3 模块调用关系	4
2.4 主要模块接口设计	5
2.4.1 用户管理模块	5
2.4.2 老人信息管理模块	6
第三章 详细设计	7
3.1 数据库 ER 图	7
3.2 核心界面流程	8
3.3 认证流程	9
第四章 测试报告	9
第五章 安装及使用	9
第六章 项目总结	10
3.1 技术难点与解决方案	10
3.2 升级方向	10
参考文献	10

第一章 需求分析

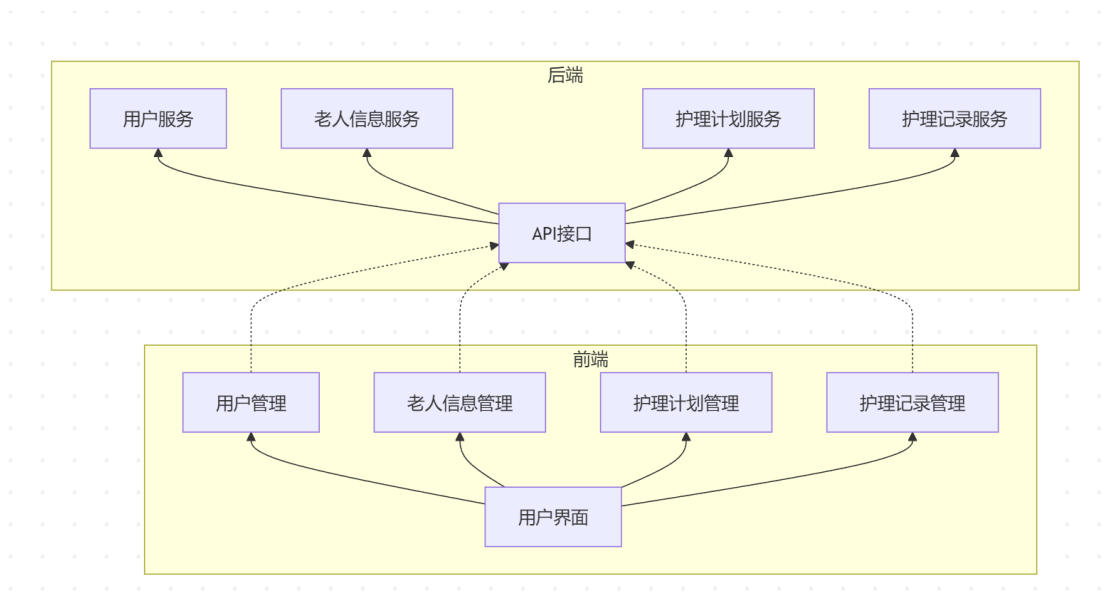
GoldenYearsManager 养老院管理系统旨在解决传统养老机构信息化程度低、管理效率不高的问题。目标用户为中小型养老机构，提供完整的老人信息管理、护理计划和记录功能。系统主要功能包括：1)用户权限管理；2)老人健康档案维护；3)护理计划制定与跟踪；4)日常护理记录。系统采用 JWT 认证保障安全性，MySQL 数据库确保数据可靠性。前端采用 Vue3 实现响应式界面，后端使用 FastAPI 提供高效 API 服务。

第二章 概要设计

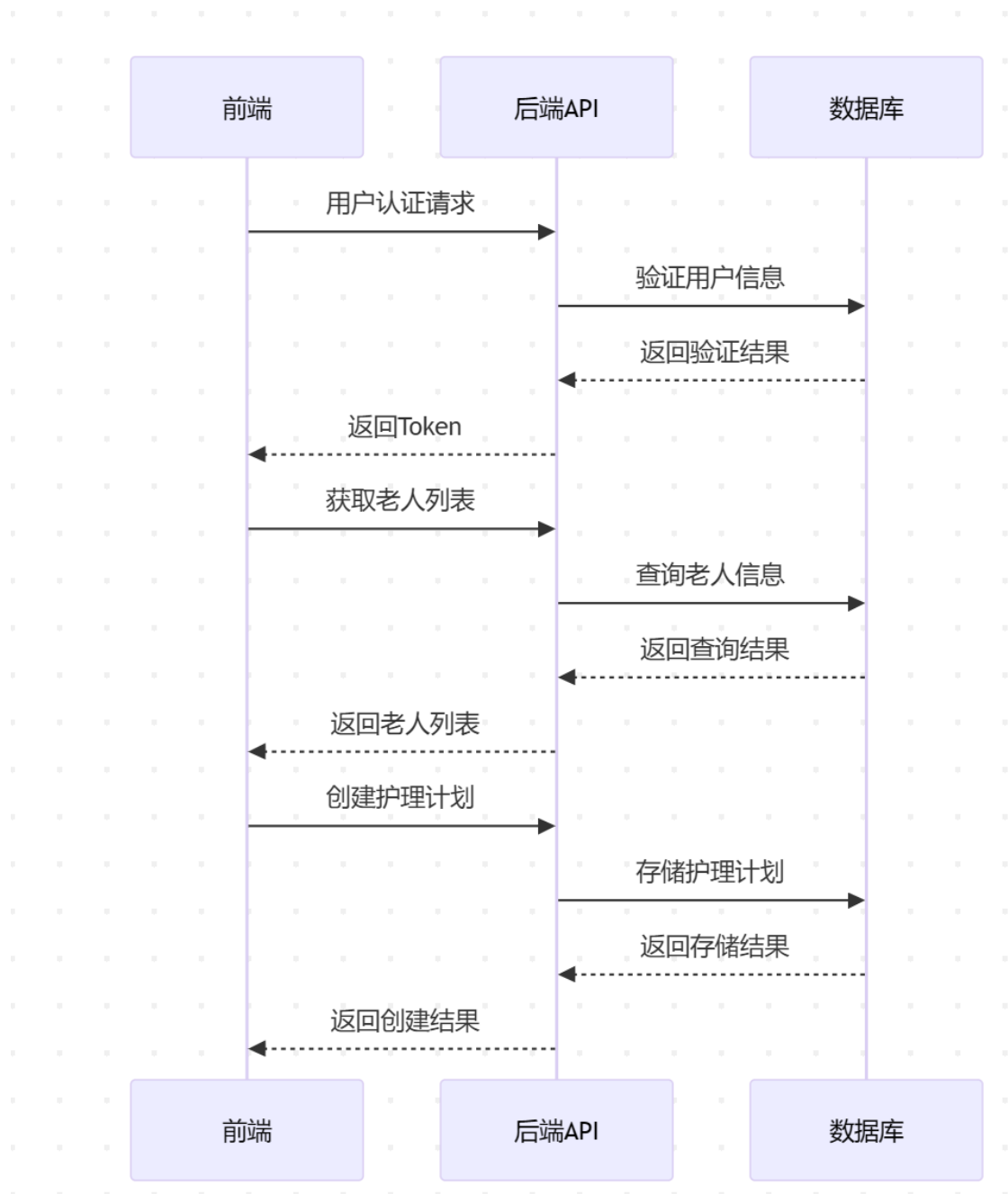
2.1 系统功能模块划分



2.2 模块层次结构

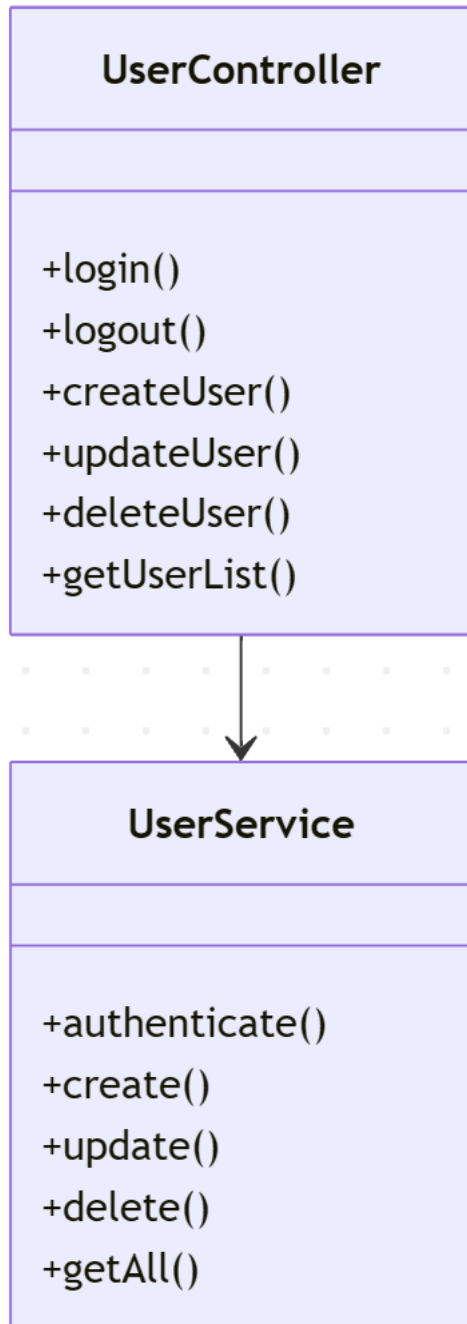


2.3 模块调用关系

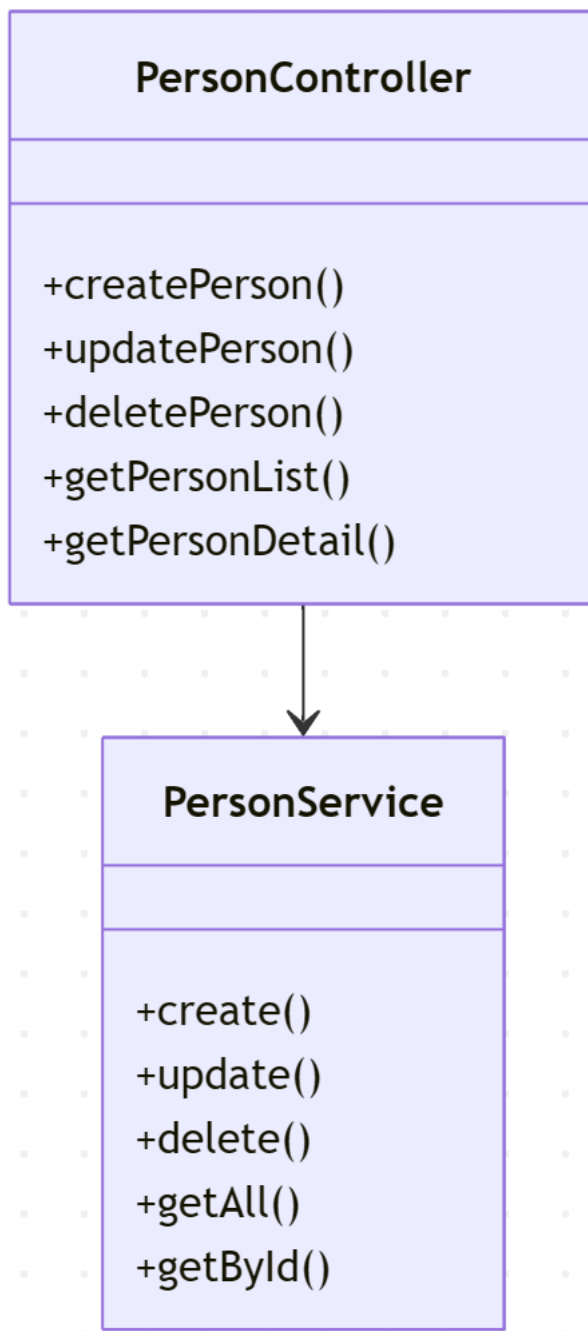


2.4 主要模块接口设计

2.4.1 用户管理模块

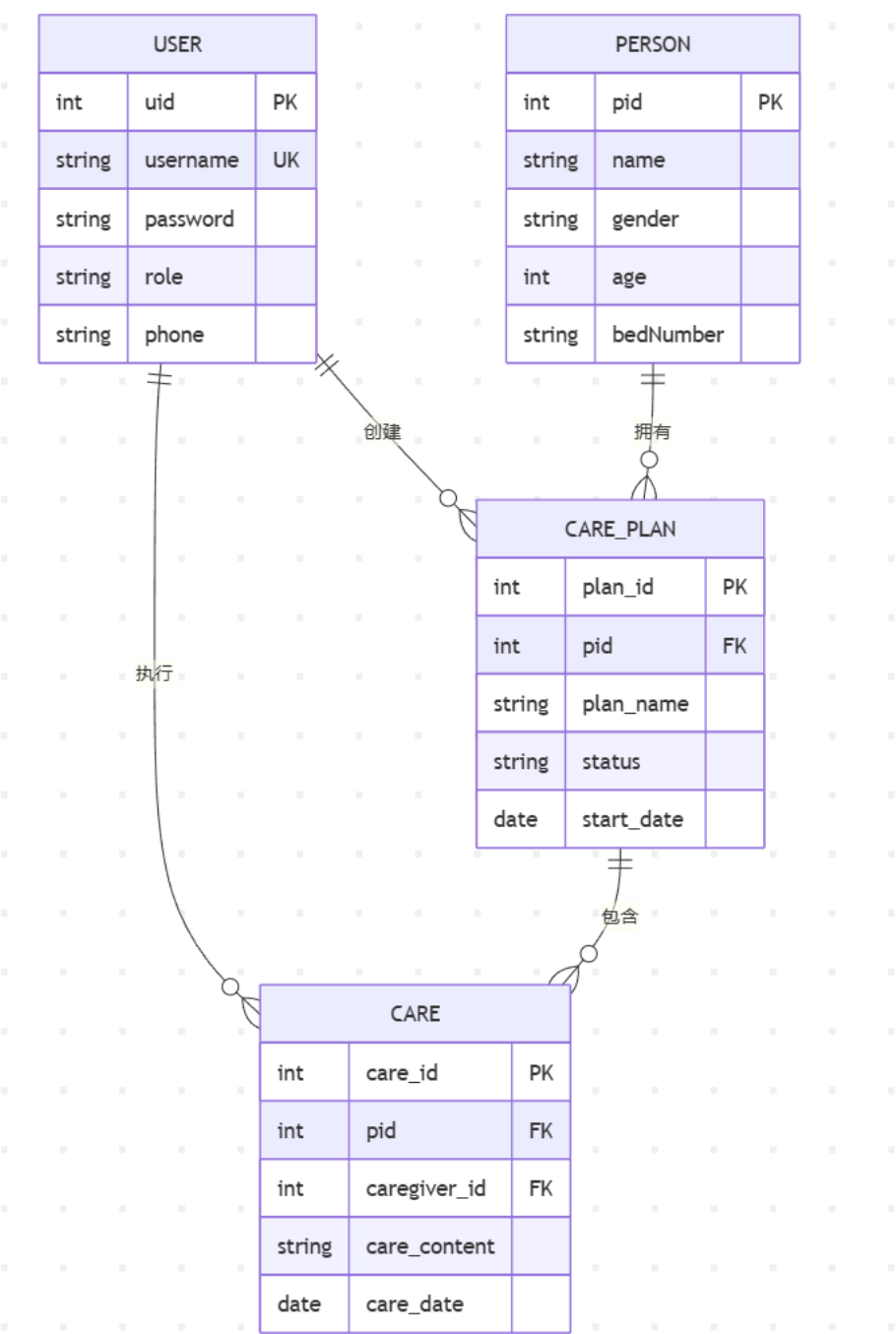


2.4.2 老人信息管理模块

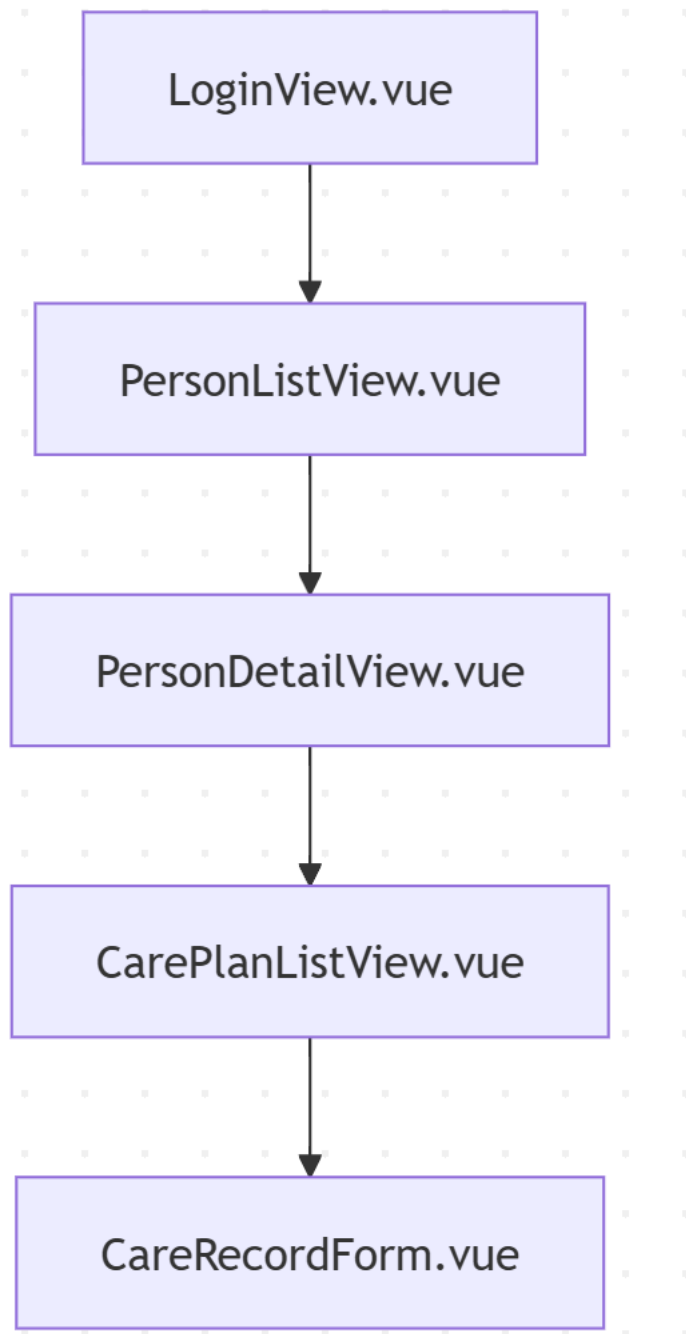


第三章 详细设计

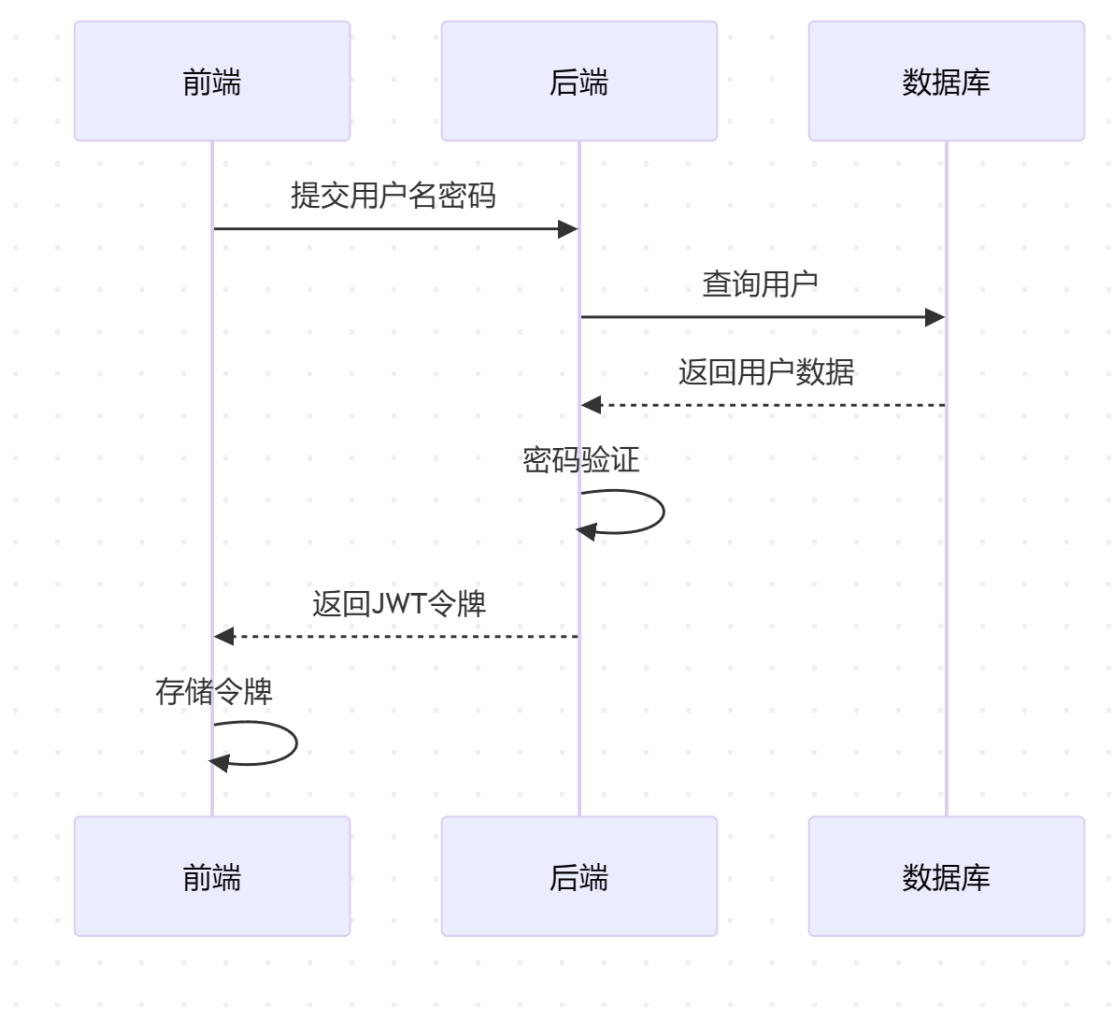
3.1 数据库 ER 图



3.2 核心界面流程



3.3 认证流程



第四章 测试报告

功能测试 – 局部输入框在输入内容后任然会提示请输入, 通过审查发现变量命名问题引发, 输入框蛇形变量, 规则小驼峰, 已修复

功能测试 – 结束时间可以早于开始时间, 原因未加规则, 加入规则, 已修复

第五章 安装及使用

将项目下载到本地, 先进入 backend 文件夹, 打开 main.py, 将 `allow_origin_regex` 的值改为 `r"^https?:/(.*)? 你的域名\. 顶级域名(:\d+)?"`

然后执行

```
docker - compose - f docker - compose.yml up -d --build
```

即可启动后端 api，然后进入 GoldenYearsManager 前端文件夹，修改 `src\api\api.ts`，将 baseUrl 改为你的 api 服务域名

```
npm install  
npm run build
```

开始构建，然后将 dist 目录的文件移动到 Apache 或 nginx 的目录下并进行配置即可

第六章 项目总结

- 采用模块化开发方式，按功能模块迭代实现
- 自定 API 规范，确保前后端一致性

3.1 技术难点与解决方案

1. 前后端联调效率
 - 痛点：单人开发需频繁切换前后端工作
 - 方案：优先后端代码，使用 Hapydev 进行 API 测试，然后根据 API 进行前端开发
2. 部署复杂度
 - 痛点：后端部署需要繁琐的设置
 - 方案：使用 Docker Compose 一键部署环境

3.2 升级方向

1. 添加出院显示
2. 添加提醒功能
3. 到计划结束时间后自动标记
4. 进行多端适配
5. 增加小程序功能
6. 打包为软件包，抛弃 web 加载以加快首次加载速度
7. 护理计划推荐
8. 异常预警
9. 数据分析报表

参考文献