

校园智能健康管理平台的设计与开发

一、项目介绍

1.1 背景与问题陈述

随着社会的飞速发展，社会环境也愈加复杂，面对复杂的社会环境，人们的心理健康问题也随之受到重视。学生是民族的未来，因此学生的心理健康问题备受各方关注。学生的心理健康问题不仅涉及到当前社会，还涉及到校园环境与学业压力等多重问题。2023年教育部等十七部门印发了《全面加强和改进新时代学生心理健康工作专项行动计划（2023—2025年）》，该计划中提出，应坚持健康第一的教学理念，切实把心理健康工作摆在更加突出位置，贯通大中小学各学段，贯穿学校、家庭、社会各方面，促进学生思想道德素质、科学文化素质和身心健康素质协调发展。

基于对教育信息化和校园安全的不断重视，一系列鼓励支持学校利用信息技术提高管理水平和服务质量政策陆续出台。智慧校园建设成为教育信息化的重要发展方向，而健康管理是智慧校园的重要组成部分。实施校园健康管理平台对于高校和学生都具有积极且长远的意义。从学校方面来说，智能化健康管理系统能提升高校学生健康管理的工作效率；从学生方面来说，有利于树立良好的健康生活方式，有效防控慢性疾病的发生。

1.2 项目目标与价值

本研究旨在设计与开发一款校园智能健康管理平台，主要以提升学生心理和身体健康管理的效率为目标。

校园健康管理平台的价值主要体现在以下几个方面：

（1）该平台致力于构建校园健康数据的动态监测体系，通过智能化技术手段完成健康信息的实时采集、精准分析与系统化管理，学生可随时记录、查看个人健康数据，从而优化校园健康管理流程，增强数据处理的时效性与科学性

（2）该平台凭借资源整合与信息共享能力，为师生提供更加全面的健康咨询与教育服务。

（3）该平台通过设置的心理测试与评估模块，助力学生掌握自己的心理状态，及时发现心理健康问题并主动求助，教师和管理人员可借助平台所给出的工具，按系统方式推进心理健康教育，加大对学生心理健康的关心与引导。

(4) 该平台为学校后续在健康干预及教育决策工作上提供了数据支持，辅助学校升级健康管理校园方案。

1.3 功能需求分析

用户端小程序主要针对学生群体，其目的是为用户提供便捷的健康管理服务，小程序有健康数据记录、心理测试模块、健康资讯检索等核心功能，健康数据记录模块允许学生把个人健康信息录入并跟踪，像身高、体重、心率、视力这类，进而借助数据可视化工具（如折线图）形成清晰的健康趋势分析；心理测试模块专门提供多种心理测评问卷，辅助学生开展自我心理状态评估，支持测试结果查询；健康资讯模块拥有分类检索的功能，学生能按照自身喜好去查阅最新的健康知识及校园公告，综合提升健康素养。

PC教师端设计聚焦于为教师提供全方位支持学生健康管理的工具，教师端具备的关键功能有学生信息管理、心理健康管理和心理试题管理，凭借学生信息管理模块，教师可查看、分析学生的健康数据以及心理测试的历史详情，为个性化指导提供数据方面的支撑；心理健康管理模块赋予教师创建和调整心理测试试卷的权限，进而设定评分规则辅助进行心理评估；心理试题管理模块支持对试题做新增、编辑或删除处理，还可快速把试题库更新，使测试内容可灵活契合不同场景及需求。

PC管理端充当平台的核心运维模块，主要面向的对象为校园管理人员，其具备的功能有知识分类、学生健康数据管理、教师与管理权限管理等，管理员凭借知识分类和科普知识管理模块，可高效发布与分类健康资讯，保证健康教育信息的及时性与精准度；学生健康数据管理模块可对全校学生健康数据进行实时汇总与趋势分析，为校园健康项目规划给予数据支撑；管理员管理模块辅助学校分配、维护教师及其他管理员的系统权限，保障系统运行的安全与规范。

1.4 技术选型与架构概述

本平台采用现代前端技术Vue、ElementUI及Uni-app构建用户端小程序，后端则基于Java、SpringBoot与MybatisPlus架构，利用Mysql数据库进行数据存储。平台分为三个主要部分：用户端小程序、PC教师端及PC管理端。

二、需求分析与设计

2.1 用户画像与场景分析

不同用户角色在业务场景的需求上各有侧重，以下将重点说明主要用户（学生、教师和管理人员）对系统的具体需求：

(1) 学生端

学生作为系统的主要用户，得具备健康数据记录、心理测试和健康资讯获取等功能。具体场景包括：

健康数据记录：学生希望能够在小程序中方便地记录日常的健康数据，包括饮食习惯、运动量、睡眠质量以及心理状态评估结果，这就需要系统具备支持数据实时更新以及历史数据可视化的能力，辅助学生清晰了解自身健康状况。

心理测试功能：学生需借助系统开展在线心理自我测评，系统应给出多种心理评估工具，采用科学的测评模型，评估结果可引导学生认清自我心理状态，必要时给出心理咨询的推荐指引。

健康资讯检索：学生希望能够快速获得健康方面的科普知识以及校园心理健康活动资讯，支持开展关键词搜索与内容推荐，按照个人兴趣情况进行个性化资讯推送。

(2) 教师端

教师在平台中主要负责学生健康管理和评估，教师端需求主要体现在以下几个方面：

学生信息管理：教师需要查询、记录和评估学生的健康数据，囊括测试结果、学年跟踪及健康变化分析，产出个性化的健康报告。

心理评估管理：教师要定期针对班级学生开展心理健康评估，系统应具备构建测试问卷、设定评分准则、汇总剖析结果的功能，由此更精准地制定有针对性的干预方案。

教学资源分享：教师期待平台能支撑教学资源的分享与交流，具备心理健康教育课程设计、课件上传等相关功能，增强教师之间的互动协作。

(3) 管理端

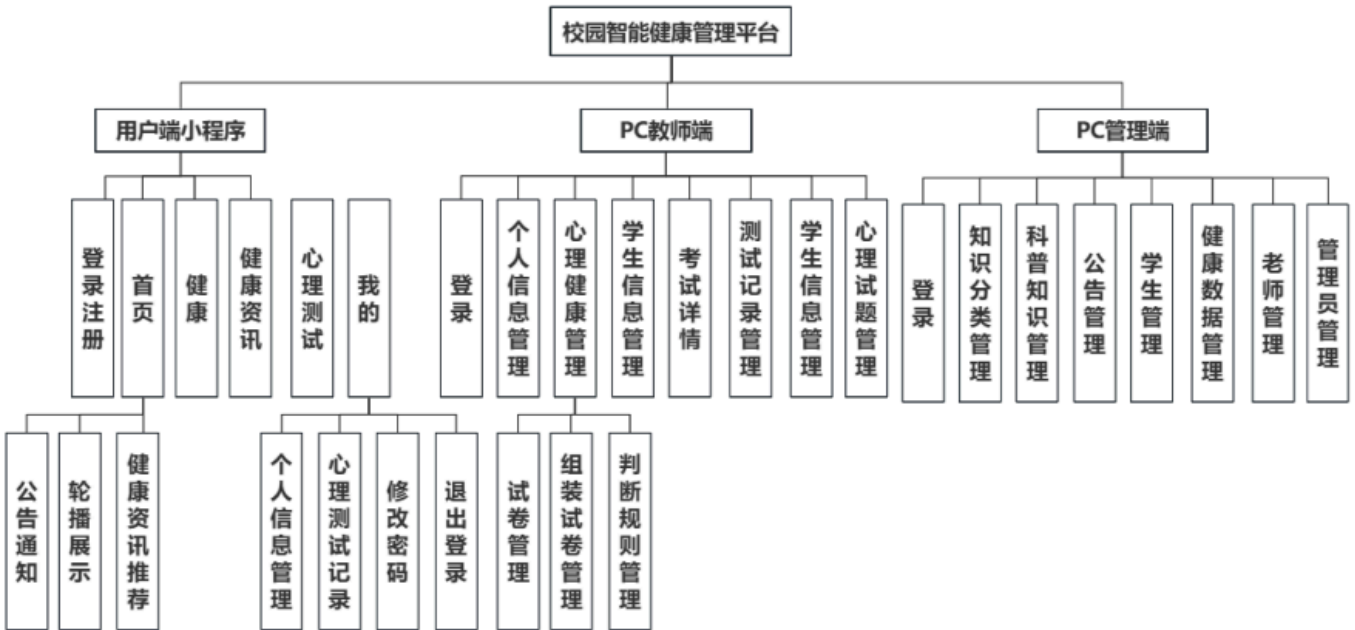
管理人员的需求相对繁杂，主要涉及平台的维护以及管理事项，具体包括：

系统监控与数据管理：管理人员需对全校学生健康数据开展统计分析，把控其趋势，用此评估学生整体健康情形，进而制订相关政策。

公告与知识库管理：管理员有权发布校园健康类的公告及活动通知，并维护健康教育相关的知识库，以此保证信息既及时又精准。

权限管理与用户管理：管理人员具有控制各个角色的访问权限，保证系统安全及数据隐私，助力合规管理实施。

基于以上分析，本系统组织架构图如下：



2.2 功能性需求

- (1) 用户管理模块：做到用户注册、登录、权限管理及数据保护功能，不同用户角色应赋予不同的权限，以达成保障数据安全的目的。
- (2) 数据记录与分析模块：学生可随时上传健康数据，教师和管理员可凭借后台对数据进行管理和解析，生成可视化报告，以便查看健康走向。
- (3) 心理测试与评估模块：给予多种心理测评工具，教师可对测试结果予以审核，再个性化设置干预手段。
- (4) 健康资讯推荐模块：系统按照用户偏好与历史举动，给学生推荐专属的健康内容和校园活动资讯。

2.3 用户体验设计

本系统需要保障实际使用时的有效性与可持续性：

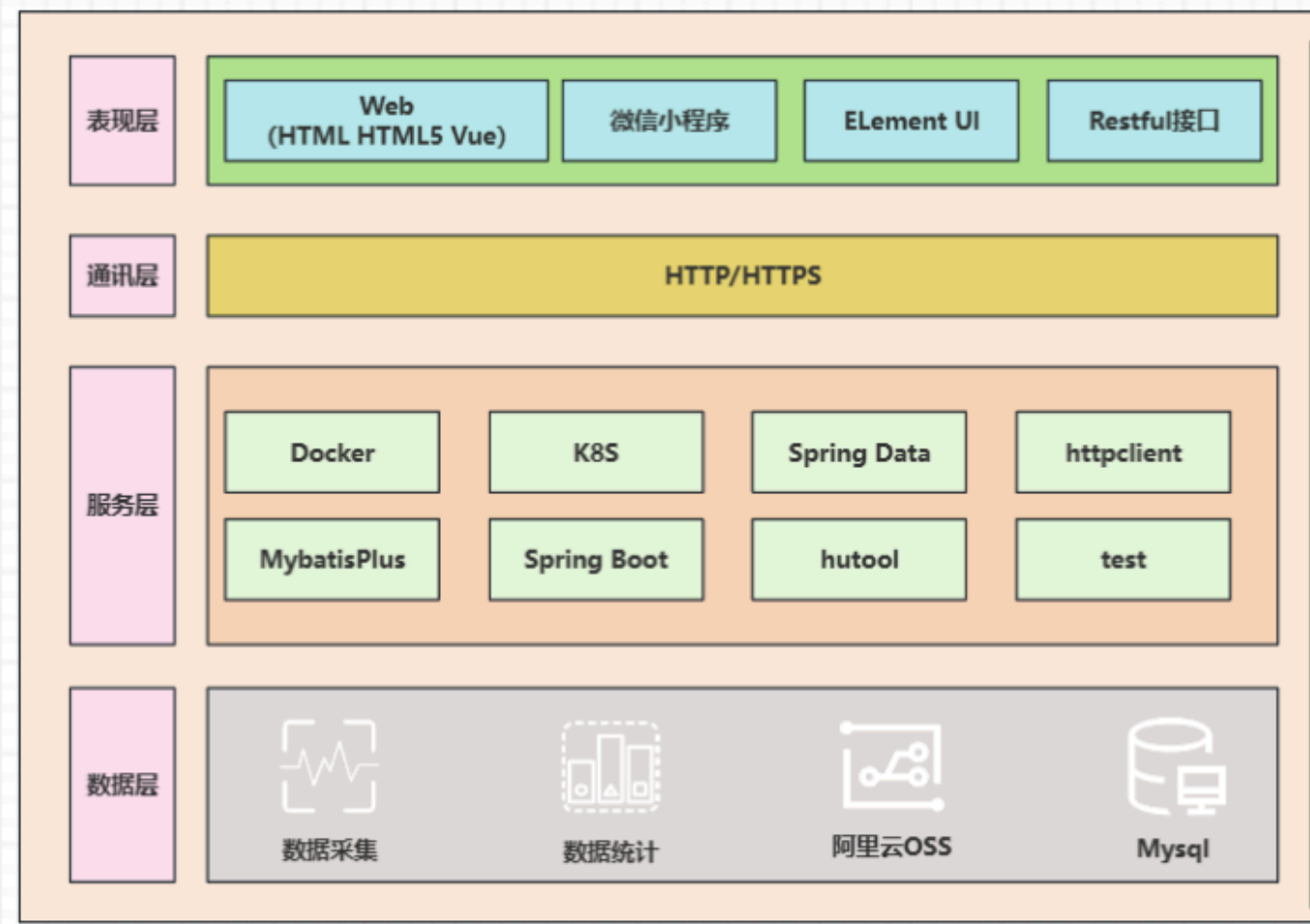
- (1) 安全性需求：系统应实现用户数据的加密存储跟传输，并实施基于角色的访问控制手段，防止敏感数据外漏。
- (2) 可用性需求：系统应做到支持高并发的用户访问，维持在线操作的顺畅效果，并且可快速排除故障，保证系统持续可用。
- (3) 可扩展性需求：系统架构设计应顾及未来功能拓展的需求，可实现更多用户和新健康管理模块的集成。

(4) 易用性需求：系统界面要简明易懂，用户利用简单导航可迅速觅得所需功能，增进用户的体验，降低学习成本支出。

三、系统架构设计

3.1 整体架构设计

系统整体技术架构图如下：



(1) 前端框架：Vue、ElementUI、Uni-app

前端部分采用Vue作为核心框架，构建模块化的组件体系，其响应式数据绑定和虚拟DOM技术显著提升了开发效率和页面渲染性能。ElementUI提供丰富的UI组件，确保了一致性且高效实现了教师端和管理端的交互功能。Uni-app支持跨平台开发，实现了用户端小程序在多终端（如微信、支付宝）上的同步发布。

(2) 后端框架：SpringBoot与MybatisPlus

SpringBoot通过其高度集成的生态系统，简化了系统配置和模块化开发。MybatisPlus作为持久层框架，提供了开发者友好的CRUD接口和自定义查询功能，极大提高了后端开发效率。

(3) 数据库：MySQL

数据存储方面，采用MySQL作为关系型数据库管理系统。MySQL具备良好的扩展性和性能支持，尤其在复杂查询场景中可以通过索引优化进行性能提升。

3.2 开发模式

(1) 前后端分离开发

系统采用前后端分离架构，后端提供标准化的RESTful API接口服务，前端调用接口实现数据交互。这种模式可以使前后端团队独立开发，提高开发效率和代码解耦程度。

(2) MVC设计模式（后端）

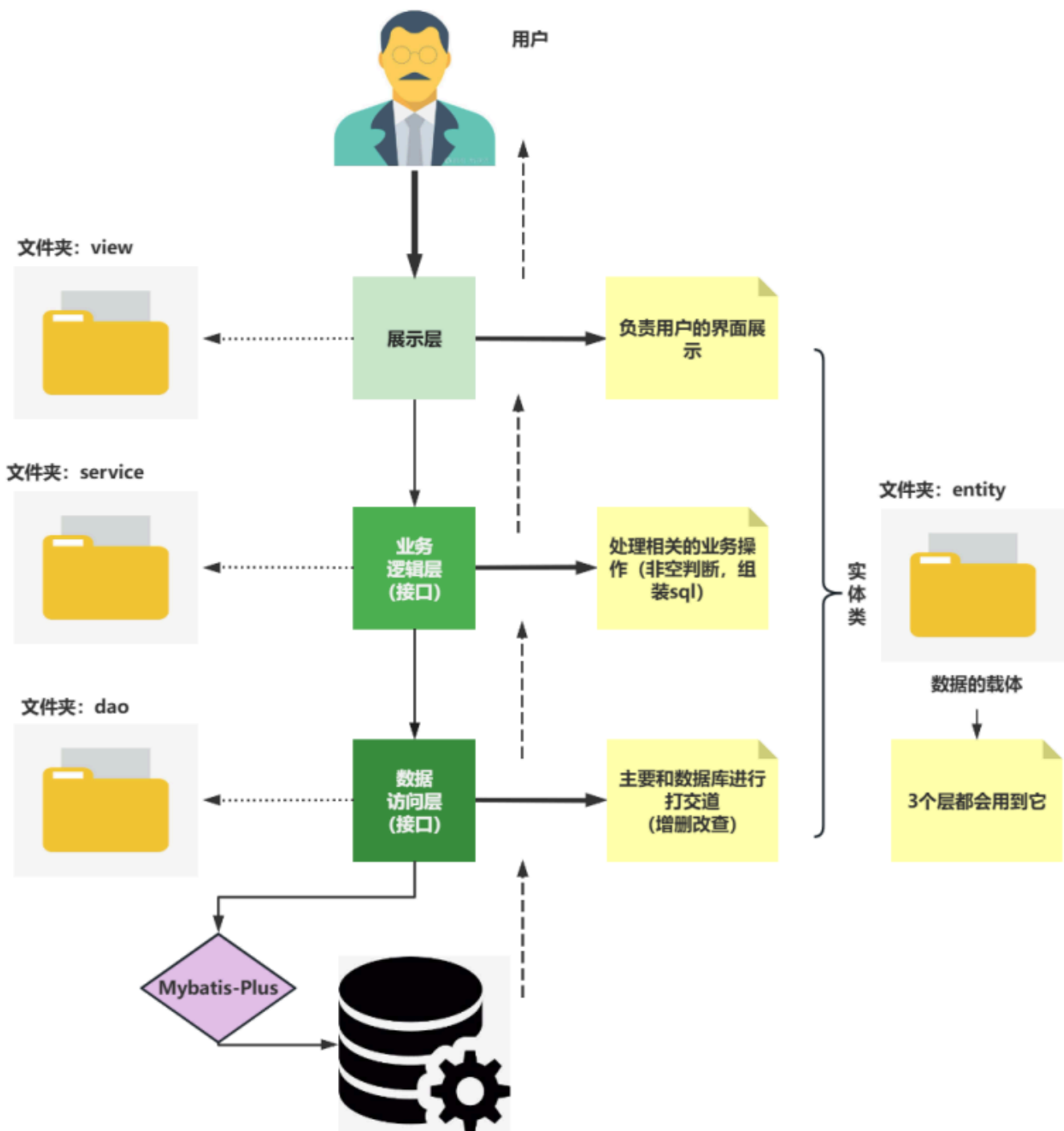
后端基于Spring框架遵循MVC模式，将系统分为三层：控制层（Controller）、业务处理层（Service）、数据访问层（Mapper），有效地降低了系统的复杂度，增强了代码的可扩展性和可维护性。

(3) 组件化开发（前端）

前端按照Vue框架的组件化思想进行开发，将系统功能模块化为多个自定义组件，每个组件控制自己的UI结构和逻辑。这种设计方式增强了代码的复用性和开发效率。

3.3 技术架构分层

校园智能健康管理平台采用分层架构的设计思路，主要由前端用户层、后端服务层和数据库层三个基本层次组成，依靠这种分层架构，可合理分离不同功能模块，使系统拥有更良好的可维护及灵活性。



3.3.1 前端用户层

包括移动端小程序和PC端教师管理系统，负责用户交互与数据展示。学生通过小程序进行健康数据录入、心理测试和健康信息查询，教师在PC端进行学生信息管理、心理健康评估等相关操作。前端界面需要设计得友好些，以此优化用户体验，确保用户能够轻松访问所有核心功能。

3.3.2 后端服务层

后端服务层以API架构为基础，利用MVC三层架构实现系统各个功能模块的独立性和可扩展性。后端负责接收前端请求、处理业务逻辑、向数据库发送查询与更新请求，最终把结果回传给前端，其主要包含的模块有：

用户管理服务：担起用户注册、登录、权限控制等功能。

健康数据服务：实施学生健康数据管理与趋势分析操作，具备数据存取及统计的支持能力。

心理测试服务：管理心理测评问卷的创建、结果分析及报告生成。

通知与资讯服务：支持把健康知识发布出去并推送校园活动。

3.3.3 数据库层

数据库层用于存储所有用户的相关数据以及系统运行必备的信息，涉及学生的健康记录、测评结果、教师管理的数据以及健康知识库。采用关系型数据库（如MySQL）或非关系型数据库（如MongoDB）由数据结构复杂性和查询需求决定，保证数据的安全和完整。

3.4 移动端设计

3.4.1 用户登录注册

用户登录注册模块是用户开启系统访问的起点，学生通过输入手机号及密码进行登录，初次运用时要完成注册，注册期间要提供个人信息，以便系统实施管理。在提供手机号基础上，用户还有权选择注册额外信息以定制体验。系统里设置了多重身份验证途径，以守护每个账户的安全性及隐私，注册及登录过程借助加密算法加密用户数据，以此防止数据泄漏和未获授权的访问。

3.4.2 健康数据管理

健康数据管理模块展示学生的健康记录和历史趋势分析。主界面设有个人信息的快速查看通道，如头像和个人健康数据的摘要。用户可查看各种健康数据类型的历史记录（如身高、体重、视力、脉搏和心率等），并通过可视化图表的方式了解其变化趋势。这些数据图表借助简洁的折线或柱状图形式予以展示，助力用户直观把握健康变化，增进自我健康管理，系统支持历史数据导出与打印这一功能，便于学生跟健康顾问还有医护人员沟通。

3.4.3 健康资讯与心理测试

健康资讯模块通过为学生提供最新健康知识和相关资讯，提高健康意识。用户可利用输入关键字或者选定资讯类别去进行信息检索，获取包含生活方式、营养、锻炼等主题范畴的文章，点击资讯马上查看详细内容；心理测试模块赋予用户进行心理健康自评测的权利，采用关键字搜索或者选择测试类别方式可参与针对性心理测试，同时保存好结果，供后续进一步参考，心理测试数据不仅供用户做自我评估，也能让健康管理专业人员给予进一步支持。

3.5 Web端设计

3.5.1 管理后台

管理员能借助浏览器访问后台，准许管理员借助专用账号进行登录，管理员账号的维护工作由平台管理团队承担，保证系统的安全可靠。

用户和健康数据管理：管理员后台赋予管理用户信息的功能，涉及学生账号的创建、删除与信息变更，系统赋予管理员查看学生健康数据和进行数据分析的权限，辅助实施健康干预。

系统监控与反馈：管理员可实时查看系统的使用情形，囊括用户活跃度和健康数据的更新情形，而且能经由仪表板获取关键指标，这样能保证平台稳定运行，迅速应对异常情形。

多功能标签页管理：后台设计囊括多个标签页，每个标签页都针对一种管理任务（好比，诸如用户管理、健康数据统计等，管理员可高效地在各个页面彼此之间切换，通过简化操作流程，提高管理水平。

3.5.2 教师端

教师端界面是专为校园教师设计的，目的是支持教师查看学生健康情况，进行教育相关的健康活动。

学生健康报告查看：教师有途径访问学生的健康记录，凭借数据可视化把握每位学生的健康变动情形，平台提供图表及报告，辅助教师识别健康风险，由此能指导学生去进行健康改善。

健康教育与通知功能：平台赋予教师发布健康相关教育内容与活动公告的权限，教师可按固定周期发送通知，经邮件或者平台消息系统对学生实施指导，该功能不但促进了学生跟教师之间的沟通，同样对健康知识的普及传播起到积极效果。

反馈和交流：教师端助力教师与学生间的健康反馈沟通互动，教师可针对这次交流为学生给出个性化健康建议，进而依据学生反馈一起改动健康计划，经过记录与分享，教师可与整个健康管理团队开展有效协作，为学生给予持续的健康援助。

3.5.3 Web 界面设计规范

本系统的Web端设计基于Element UI，主界面布局如图3.2所示。Element UI由饿了么前端团队开发，是一款基于Vue.js的广受欢迎的桌面UI框架，以其简洁优雅的设计风格和易用性深受开发者喜爱。其设计理念涵盖四大核心原则：一致性、反馈性、高效性和可控性。一致性体现在遵循用户习惯并保持界面元素与结构统一；反馈性通过特殊样式和动态效果明确展现操作与变化；高效性旨在简化流程，降低用户使用和学习的难度；可控性则为用户决策提供提示与建议，避免强制代替用户做出选择。

Web端主页面布局图如下：



四、API设计

4.1 API 设计原则

4.1.1 RESTful API 设计原则

1. 资源抽象：将校园智能健康管理平台的功能抽象为资源，例如学生心理健康档案（/students/{studentId}/mental-health-records）、心理测评结果（/assessments/{assessmentId}/results）。每个资源都有唯一的 URI 进行标识，以方便客户端进行访问和操作。
2. 统一接口：使用标准的 HTTP 方法来定义对资源的操作。GET用于获取资源，比如GET /students/{studentId}获取某个学生的信息；POST用于创建新资源，像POST /assessments创建一条新的心理测评记录；PUT用于更新资源的全部内容；PATCH用于部分更新；DELETE用于删除资源，如DELETE /students/{studentId}/mental-health-records/{recordId}删除学生的某条心理健康记录。
3. 无状态通信：服务器不保存客户端的状态信息，每个请求都包含了处理该请求所需的全部信息。这样可以提高系统的可扩展性和可靠性，即使某个服务器节点出现故

障，其他节点也能根据请求信息正常处理请求。

4. 分层系统：整个 API 系统分为多个层次，包括客户端、API 网关、业务逻辑层和数据存储层。客户端通过 API 网关访问业务逻辑层，API 网关负责统一接入、认证授权、流量控制等功能；业务逻辑层处理具体的资源操作和业务逻辑；数据存储层负责数据的持久化和访问，各层之间职责明确，降低系统的耦合度。

4.1.2 命名规范

1. 资源命名：采用复数名词来表示资源集合，例如/students表示学生资源集合，/assessments表示心理测评资源集合。对于单个资源，则通过在 URI 中添加唯一标识符来定位，如/students/{studentId}，其中{studentId}是学生的唯一标识。
2. 子资源命名：如果资源存在层级关系，使用斜杠来分隔子资源。例如/students/{studentId}/courses/{courseId}/assessments，表示获取某个学生在某门课程下的心理测评记录，清晰地展示了资源之间的关联。
3. 避免动词：不在 URI 中使用动词，而是通过 HTTP 方法来表达对资源的操作。例如，不使用/createStudent，而是使用POST /students来创建学生资源，遵循 RESTful 规范，使 API 更加简洁和统一。
4. 参数命名：使用驼峰命名法来命名查询参数，例如/students?pageSize=10&pageNumber=2，其中pageSize表示每页显示的数量，pageNumber表示页码。常用的参数如sort用于指定排序字段，filter用于添加过滤条件，方便客户端根据需求获取特定的数据。

4.1.3 版本控制策略

1. URI 版本控制：在 URI 中包含版本号，如/api/v1/students、/api/v2/students。这种方式的优点是实现简单，客户端可以通过明确的版本号来访问不同版本的 API，兼容性好，也便于进行缓存隔离；缺点是会污染 URL，并且需要维护多个版本的代码。适用于 API 发生大规模变更，或者有较多第三方集成的场景。
2. 请求头版本控制：在请求头中指定 API 版本，例如Accept: application/vnd.psychology.v2+json，表示客户端期望获取版本 2 的 JSON 格式数据。该方法的优点是能够保持 URI 的纯净，支持内容协商；缺点是客户端需要显式指定版本，并且缓存管理相对复杂。比较适合内部服务调用、微服务架构的场景。
3. 查询参数版本控制：通过在查询参数中添加版本号来进行控制，如/students?version=v2。这种方式实现简单、灵活，但它违反了 REST 原则，并且容易导致缓存失效，一般仅作为过渡方案使用。

4.3 接口文档

由于系统采用前后端分离的架构，接口设计旨在减少同一功能或一组功能调用多个接口的情况，从而确保接口的简洁性和易用性。本系统使用RESTful风格进行接口设计，并通过Postman进行接口测试。

在请求类型方面，根据不同接口的功能，采用了POST、GET、PUT和DELETE四种类型，以确保各API接口的安全性和可靠性。除登录和注册接口外，其他大部分接口均要求请求头中携带token字段，以进行身份认证。

在数据传递方式上，依据数据量 and 安全性要求的不同，选择最合适的携带方式。对于数据量较少且安全性要求较低的接口，使用params参数传值；而在数据量较大或安全性要求较高的接口中，则通过请求体传递数据，主要采用raw方式下的application/json格式。此外，文件上传接口使用form表单的请求体提交数据。

响应体采用统一封装格式，设计为 {code, msg, data}。其中，code表示状态码，能够根据不同寻常类型返回相应的状态码；msg用于存放描述信息，包含异常信息；data持有业务实体，返回的信息可以是单个字符串、JSON数据，或者是JSON列表。

4.3.1 PC端接口

序号	接口地址	请求方法	接口描述
1	/testPaper/page	POST	试卷分页查询
2	/testPaper/save/update	POST	添加/修改试卷
3	/testPaper/get/{id}	GET	根据ID获取试卷
4	/testPaper/remove	POST	删除试卷
5	/testQuestions/page	POST	试题分页查询
6	/testQuestions/save/update	POST	添加/修改试题
7	/testQuestions/get/{id}	GET	根据ID获取试题
8	/testQuestions/remove	POST	删除试题
9	/testQuestions/all/testPaper/{id}	GET	获取指定试卷所有试题
10	/student/page	POST	学生分页查询
11	/student/save/update	POST	添加/修改学生信息
12	/student/get/{id}	GET	根据ID获取学生信息
13	/student/remove	POST	删除学生信息
14	/teachers/page	POST	教师分页查询

序号	接口地址	请求方法	接口描述
15	/teachers/save/update	POST	添加/修改教师信息
16	/teachers/get/{id}	GET	根据ID获取教师信息
17	/teachers/remove	POST	删除教师信息
18	/questions/page	POST	心理试题分页查询
19	/questions/save/update	POST	添加/修改心理试题
20	/questions/get/{id}	GET	根据ID获取心理试题
21	/questions/remove	POST	删除心理试题
22	/records/page	POST	测试记录分页查询
23	/records/save/update	POST	添加/修改测试记录
24	/records/get/{id}	GET	根据ID获取测试记录
25	/records/remove	POST	删除测试记录
26	/manage/login/tea	POST	教师登录
27	/manage/login	GET	获取登录信息
28	/manage/login/logout	GET	用户登出
29	/healthy/echarts/{id}	GET	获取健康图表数据
30	/upload/file	POST	文件上传接口

4.3.2 移动端接口

序号	接口地址	请求方法	接口描述
1	/notice/findList	POST	公告列表查询
2	/knowledge/getIndex/{count}	GET	首页最新资讯查询
3	/knowledge/findList	POST	知识库多条件查询
4	/types/findList	POST	知识分类查询
5	/testPaper/findList	POST	试卷列表查询
6	/student/login	POST	学生登录
7	/student/save/update	POST	学生信息保存/更新
8	/student/findList	POST	学生信息查询
9	/testQuestions/allQuestion	POST	获取试卷全部题目
10	/records/saveGrade	POST	考试成绩保存

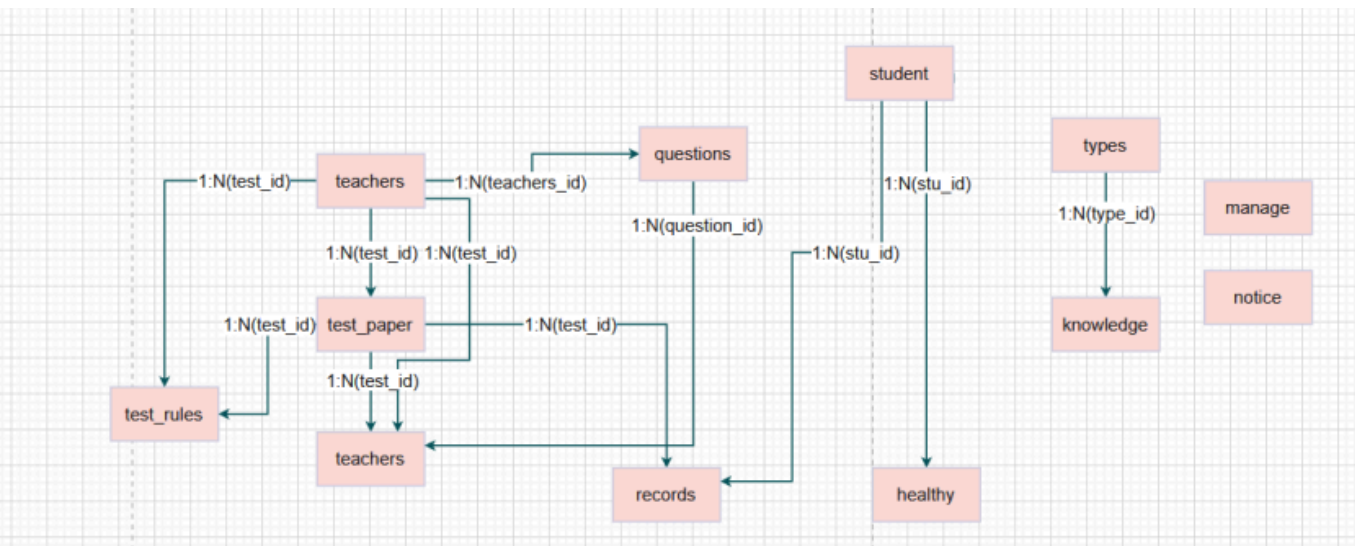
序号	接口地址	请求方法	接口描述
11	/records/findList	POST	考试记录查询
12	/teachers/findList	POST	教师列表查询
13	/healthy/echarts1/{userId}	GET	健康数据图表查询

五、数据库设计

5.1 数据模型设计

在每一个系统中数据库有着非常重要的作用，数据库的设计得好将会增加系统的效率以及系统各逻辑功能的实现。所以数据库的设计要从系统的实际需要出发，才能使其更为完美的符合系统功能的实现。

概念模型是对现实中的问题出现的事物的进行描述，ER图是由实体及其关系构成的图，通过E-R图可以清楚地描述系统涉及到的实体之间的相互关系。E-R设计图如下：



主数据库选用 MySQL（关系型数据库），MySQL 作为主数据库管理系统，负责存储结构化核心业务数据。其成熟稳定的特性、完善的社区支持以及强大的事务处理能力（ACID），使其成为管理用户信息、权限关系、心理健康档案等数据的理想选择。MySQL 丰富的索引类型（如 B-Tree、哈希索引）能优化多条件查询，与 MyBatis-Plus 框架的无缝集成则降低了开发成本。

辅助数据库选用 MongoDB（文档数据库），MongoDB 的灵活文档模型和强大的查询能力，使其非常适合存储半结构化数据。对于心理测评结果和咨询会话记录等数据，MongoDB 无需预定义模式，可轻松应对数据结构的动态变化。其内置的聚合框架支持复杂数据分析，水平扩展能力也能满足未来数据量增长的需求。

5.2 数据库架构

根据E-R得出数据库包涵了以下几张数据表来实现了数据库的存储、调用。

healthy（健康数据表）此表记录学生的健康信息，包括身高、体重、视力和多项健康指标，以及AI生成的健康建议和录入时间，供健康管理分析和历史记录留存。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	数据编号
2	stu_id	bigint	20	否	学生编号
3	height	double	-	否	身高
4	weight	double	-	否	体重
5	eyesight	varchar	255	否	视力
6	ulse_rate	varchar	255	否	脉率
7	blood_pressure	varchar	255	否	血压
8	heart_rate	varchar	255	否	心率
9	breath	varchar	255	否	呼吸
10	blood_sugar	varchar	255	否	血糖
11	blood_lipid	varchar	255	否	血脂
12	month_day	date	-	否	月份
13	create_time	datetime	-	否	录入时间
14	content	text	-	否	AI建议

knowledge（科普知识管理表）此表记录健康相关的科普知识内容，包括所属分类、标题、封面图片及详细内容，用于在平台上展示健康知识。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	科普知识编号
2	banner_img	varchar	255	否	封面图片

序号	列名	数据类型	长度	主键	说明
3	type_id	bigint	20	否	所属分类
4	content	text	-	否	内容
5	info	varchar	255	否	简介
6	create_time	datetime	-	否	创建时间
7	title	varchar	255	否	标题

manage（管理员表）此表存储管理员的账号信息，包括用户名、密码、照片及姓名，用于管理员的系统身份验证与功能操作。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	管理员编号
2	user_name	varchar	50	否	用户名
3	pass_word	varchar	200	否	密码
4	photo_img	varchar	255	否	图片
5	name	varchar	50	否	名称

notice（公告管理表）此表记录校园平台上的公告信息，包括标题、内容、封面图片以及公告状态，用于在系统中展示。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	公告编号
2	title	varchar	255	否	标题
3	notice_img	varchar	255	否	封面图片
4	content	text	-	否	内容
5	info	varchar	255	否	简介
6	status	int	11	否	状态[0：隐藏，1：正常]
7	create_time	datetime	-	否	创建时间

questions（题目表）此表存储心理健康测试中的题目内容及选项，包含每个选项的分数，用于健康测试模块。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	题目编号
2	title	text	-	否	题目标题
3	option_a	varchar	255	否	选项A
4	option_b	varchar	255	否	选项B
5	option_c	varchar	255	否	选项C
6	option_d	varchar	255	否	选项D
7	grade_a	varchar	255	否	答案A分数
8	grade_b	varchar	255	否	答案B分数
9	grade_c	varchar	255	否	答案C分数
10	grade_d	varchar	255	否	答案D分数
11	teacher_id	bigint	20	否	老师编号
12	create_time	datetime	-	否	创建时间

records（建档信息表）此表记录学生的测试档案信息，包括学生编号、试卷编号、测试成绩以及对应的评估结果，供健康管理参考。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	档案编号
2	stu_id	bigint	20	否	学生编号
3	test_id	bigint	20	否	试卷编号
4	tea_id	bigint	20	否	教师编号
5	grade	varchar	255	否	考试分数
6	create_time	datetime	-	否	创建时间
7	info	varchar	255	否	对应结果

student（学生表）此表存储学生的基本信息，包括姓名、年龄、性别、联系方式等，用于标识和管理学生用户。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	主键ID
2	user_name	varchar	100	否	用户名（select）

序号	列名	数据类型	长度	主键	说明
3	pass_word	varchar	255	否	密码
4	head_img	varchar	255	否	头像 (img)
5	nick_name	varchar	100	否	姓名
6	create_time	datetime	-	否	创建时间

teachers（教师表）此表存储教师的个人信息和系统使用信息，包括姓名、账号、联系方式等，用于教师管理和登录功能。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	教师编号
2	username	varchar	255	否	账号
3	password	varchar	255	否	密码
4	tea_name	varchar	255	否	姓名
5	tea_age	int	20	否	年龄
6	tea_sex	int	20	否	性别
7	tea_info	text	-	否	简介
8	tea_tel	varchar	255	否	联系方式
9	tea_img	varchar	255	否	头像
10	create_time	datetime	-	否	创建时间
11	mail	varchar	255	否	邮箱
12	status	int	11	否	状态[0：待审核，1：可用，2：停用]

test_paper（试卷管理表）此表存储与健康测评相关的试卷信息，包括标题、封面图片和总分等内容，供教师创建和学生使用。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	试卷编号
2	tea_id	bigint	20	否	所属老师
3	title	varchar	255	否	试卷标题
4	img	varchar	255	否	封面图片
5	create_time	datetime	-	否	创建时间

序号	列名	数据类型	长度	主键	说明
6	total_grade	varchar	255	否	总分
7	info	varchar	255	否	试卷简介

test_questions（试卷题目表）此表将试卷与具体的题目关联，包含试卷编号、题目编号以及教师编号。

序号	列名	数据类型	长度	主键	说明
	id	bigint	20	是	主键ID
2	test_id	bigint	20	否	所属试卷
3	question_id	bigint	20	否	题目编号
4	tea_id	bigint	20	否	老师编号

test_rules（试卷规则管理表）此表存储与试卷测评规则相关的信息，包括分数区间及对应评估结果，用于学生成绩的自动评价。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	规则编号
2	test_id	bigint	20	否	试卷编号
3	min_grade	int	10	否	最低分数
4	max_grade	int	10	否	最高分数
5	info	varchar	255	否	对应结果
6	create_time	datetime	-	否	创建时间
7	tea_id	bigint	20	否	老师编号

types（科普知识分类表）此表记录健康知识的分类信息，包括分类名称和创建时间，用于知识模块数据的归类。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	分类编号
2	type_name	varchar	255	否	分类名称
3	create_time	datetime	-	否	创建时间

六、前端实现

6.1 技术栈与开发环境

在校园智能健康管理平台的前端开发中，技术栈的选择、开发工具的运用以及构建流程的设计，对项目的开发效率、运行性能及可维护性起着决定性作用。

从技术栈层面来看，本项目采用 uni-app 作为核心框架，它基于 Vue.js 语法规则，能够实现一套代码多端发布，适配微信小程序、H5、APP 等多个平台，极大降低了多端开发的成本与维护难度。UI 组件库选用 uView UI，其提供丰富且美观的组件，如表单组件、弹窗组件等，可快速搭建出符合用户操作习惯的界面，同时具备良好的响应式设计能力，确保在不同尺寸设备上都能呈现出优质的视觉效果。在状态管理方面，引入 Vuex，方便管理全局状态，使数据在组件间的传递更加清晰有序，尤其适用于处理用户登录状态、心理健康数据缓存等场景。数据请求采用 axios，它具备强大的拦截器功能，可统一处理请求头设置、错误响应等操作，同时支持 Promise，便于与 async/await 结合，实现简洁高效的异步数据请求。

开发工具的选择直接影响开发效率与代码质量。Visual Studio Code（简称 VS Code）凭借其丰富的插件生态，成为主力开发编辑器。搭配 Vetur 插件，能实现对 Vue 代码的语法高亮、智能提示、格式化等功能，提升代码编写的流畅度；ESLint 插件则按照团队统一的代码规范对代码进行实时检查，及时发现潜在错误与不规范写法，保证代码质量与团队协作的规范性；Prettier 插件自动格式化代码，统一代码风格，使代码更易阅读与维护。此外，Chrome 浏览器作为调试利器，其开发者工具提供强大的调试功能，可对页面性能进行分析、查看网络请求、调试 JavaScript 代码等，帮助开发者快速定位并解决问题。

在构建流程上，项目采用自动化构建工具实现高效开发。借助 uni-app 内置的编译工具，在开发阶段，通过 `npm run dev` 命令启动本地开发服务器，实现热更新功能，开发者修改代码后页面可实时刷新，显著提升开发效率；在打包阶段，使用 `npm run build` 命令，根据不同平台需求生成对应的代码包，如微信小程序代码包、H5 页面代码包等。构建过程中，会对代码进行压缩混淆处理，减小文件体积，提高页面加载速度；同时，对静态资源进行优化，如图片压缩、字体文件裁剪等，进一步提升应用性能。最后，将构建好的代码包部署到相应的服务器环境，完成项目的上线工作。

6.2 核心功能模块实现

(1) 异步处理：异步处理可以确保用户界面在等待后台任务完成时仍然保持响应，在 uniapp 中实现了 ECMAScript 6 (ES6) Promise 的基本功能完成异步处理的工作。

`Promise` 构造函数：用于创建新的Promise实例。

`.then()` 方法：用于注册成功和失败的回调函数。

`.catch()` 方法：用于注册失败的回调函数。

`.all()` 方法：用于等待多个Promise实例都完成。

`.race()` 方法：用于等待多个Promise实例中第一个完成的。

`.resolve()` 方法：用于创建一个立即Fulfilled的Promise实例。

`.reject()` 方法：用于创建一个立即Rejected的Promise实例。

(2) API端点管理：这段代码的目的是为了提供一个中心化的API端点管理，使得在代码中引用和修改API端点变得更加方便和统一。这样，如果服务器的IP地址或端口号发生变化，只需要修改SERVER_HOST常量即可，而不需要在代码中逐一查找和替换所有的API端点。

server-api.js

```
1  const SERVER_HOST = "http://127.0.0.1:8888"
2
3  const APIURL = {
4    getNotice: SERVER_HOST.concat('/notice/findList'), // 获取公告
5    getIndexData: SERVER_HOST.concat('/knowledge/getIndex/'), // 首页最新资讯
6    knowledgeFindList: SERVER_HOST.concat('/knowledge/findList'),
7    knowledgeTypeFindList: SERVER_HOST.concat('/types/findList'),
8    getTestPaper: SERVER_HOST.concat('/testPaper/findList'),
9    usersLogin: SERVER_HOST.concat('/student/login'),
10   saveUser: SERVER_HOST.concat('/student/save/update'),
11   usersFindList: SERVER_HOST.concat('/student/findList'),
12   getQuestion: SERVER_HOST.concat('/testQuestions/allQuestion'),
13   addGrade: SERVER_HOST.concat('/records/saveGrade'),
14   records: SERVER_HOST.concat('/records/findList'),
15   getTeacherList: SERVER_HOST.concat('/teachers/findList'),
16   getEcharts: SERVER_HOST.concat('/healthy/echarts1/'),
17 }
18
19 module.exports.APIURL = APIURL;
```

(3) 跨平台：这段代码提供了一系列工具函数，用于处理与系统信息、数据验证、字符串和数字处理相关的任务。这些函数可以用于提高代码的可读性和可维护性，同时确保在不同平台和不同情况下都能得到正确的处理结果。

```

1 function matchPhoneType() {
2   switch (uni.getSystemInfoSync().platform) {
3     case 'android':
4       console.log('运行Android上')
5       return 1;
6       break;
7     case 'ios':
8       console.log('运行iOS上')
9       return 2;
10      break;
11     default:
12       console.log('运行在开发工具上')
13       return 1;
14       break;
15   }
16 }
17
18 // 统一获取系统信息
19 function uniqueGetSystemInfo() {
20   var info = getApp().globalData.systemInfo
21   if (info == null) {
22     var res = uni.getSystemInfoSync() // 同步获取
23     getApp().globalData.systemInfo = res // App全局临时赋值
24     return res
25   } else {
26     return info
27   }
28 }
29
30
31 function getTopBarHeight(){
32   let res = uniqueGetSystemInfo()
33   let navbarHeight = (res.navigationBarHeight || res.titleBarHeight || 44)
34   return res.statusBarHeight + navbarHeight
35 }
36
37
38 // 校验手机号码是否合法
39 function isPhoneNumber(mobile) {

```

(4) 封装Http：这段代码提供了一套封装的HTTP请求处理机制，它使用Promise来简化异步操作的处理，并提供了统一的错误处理和结果处理逻辑。

```

login-mixins.js  myCache.js  page-mixins.js  request-util.js  server-api.js
1 // 这里引用es6的promise模块
2 var promise = require("utils/es6-promise.js");
3 import myCache from '@utils/myCache.js'
4 /**
5  * @author azhou
6  * @params fn 所需调用的方法
7  * @return 返回一个promise对象
8  * @备注 详情请了解es6下的promise模块
9  * @参考链接 http://es6.ruanyifeng.com/?spm=a2c4e.11153940.blogcont325262.8.573c367b88G2hG#docs/promise%E5%B4%A7%E7%94%A8
10 */
11 function myPromise(fn) {
12   return function(obj = {}) {
13     return new Promise((resolve, reject) => {
14       obj.success = function(res) {
15         console.log(res)
16         // 此处需要在后端统一返回状态码，若为0则请求成功，失败则返回其他错误信息
17         // 在方法调用时在请求方法后添加catch代码
18         if (res.code == 404) {
19           console.log("后端接口找不到了!!!")
20           return reject();
21         }
22         if (res.data.data == null) {
23           res.data.data = {}
24         }
25         if (res.data.code > 10000) {
26           return reject();
27         }
28         if (res.data.code != 200) {
29           uni.showToast({
30             title: res.data.msg,
31             icon: 'none'
32           })
33           return reject();
34         }
35         return resolve(res.data);
36       };
37     });
38   };
39   obj.fail = function(err) {
40     // 调用失败
41     if (err.msg == null || err.msg == '' || err.msg == 'SUCCESS') {

```

七、后端实现

7.1 技术栈与架构

校园智能健康管理平台后端的搭建，需综合考量系统的功能需求、性能要求以及未来的拓展性，因此在技术栈和框架的选择上需谨慎权衡。

本项目后端技术栈以 Java 为核心开发语言，选择 Spring Boot 作为基础框架，构建高效稳定的后端服务。数据库层面，采用 MySQL 作为关系型数据库，MongoDB 处理半结构化数据，Redis 实现数据缓存加速，InfluxDB 存储时序数据，形成多数据库协同的架构。在持久层操作上，引入 MyBatis-Plus 框架简化数据库操作，日志处理使用 Logback 进行规范记录，接口文档采用 Swagger 自动生成，提升前后端协作效率。

Spring Boot 框架的选用基于多方面的考量。其一，它具备强大的自动配置功能，能够快速整合各类开发场景所需的依赖，如 Web 开发、数据库连接、安全配置等，极大缩短项目搭建周期，让开发人员专注于业务逻辑的实现。其二，Spring Boot 内置的嵌入式 Web 服务器（如 Tomcat、Jetty），使得应用程序可以便捷地打包为独立运行的可执行文件，方便部署与运维。其三，该框架拥有庞大的社区生态，丰富的插件和成熟的解决方案，为项目开发过程中遇到的各类问题提供了有力支持。在校园智能健康管理平台中，Spring Boot 能够高效支撑用户管理、心理健康档案存储、测评数据处理等核心业务模块的开发，确保系统的稳定性和扩展性。

数据库选型方面，MySQL 作为关系型数据库，凭借其成熟稳定的特性、完善的事务处理机制（ACID）以及丰富的索引类型，能够妥善管理用户信息、权限配置等结构化数据，满足业务对数据一致性和复杂查询的需求。MongoDB 则以灵活的文档模型，适配心理测评结果、咨询会话记录等半结构化数据，其强大的聚合查询能力和水平扩展特性，可轻松应对数据结构的动态变化与数据量的增长。Redis 基于内存的高速读写性能和丰富的数据结构，用于缓存高频访问数据，有效提升系统响应速度。InfluxDB 专为时序数据设计，具备高效的写入性能和时间窗口查询能力，适用于存储学生心理健康指标变化等时序性数据，节省存储空间的同时提升分析效率。

MyBatis-Plus 作为持久层框架，在简化数据库操作上优势显著。它在 MyBatis 的基础上进行增强，提供了强大的代码生成器，能够根据数据库表结构自动生成基础的 CRUD 代码，减少重复开发工作；其内置的通用 Mapper 和 Wrapper 条件构造器，使数据操作更加便捷高效，开发人员无需编写大量 SQL 语句即可完成复杂的数据查询与更新操作，提升开发效率和代码质量。

Logback 用于日志处理，它提供了灵活的日志级别控制和日志格式定义，能够根据不同的运行环境和需求记录详细程度各异的日志信息，便于开发人员排查问题和分析系统运行状态。Swagger 自动生成接口文档，能够实时反映后端接口的变化，通过可视

化的界面展示接口的请求参数、响应结果等信息，有效降低前后端沟通成本，提升协作效率。

7.2 核心业务模块实现

(1) 定义数据模型 (Entity)：在Spring Boot后端，使用Java类来定义数据模型，这些实体类对应数据库中的表。使用Mybatis或者XML配置文件来映射Java类到数据库表中

(2) 编写业务逻辑层：在控制器中，编写业务逻辑来处理请求，包括数据的验证、业务规则的执行等。使用服务层 (Service) 来封装业务逻辑，使用数据访问层 (Repository或Mapper) 来与数据库交互。

(3) 编写控制层：在Spring Boot中，使用Spring MVC创建RESTful API端点，这些端点用于处理前端的HTTP请求。使用@RestController注解来创建控制器，使用@RequestMapping或@GetMapping、@PostMapping等注解来定义端点的路径和请求类型。

(4) 跨域处理：CORS是一种机制，它允许Web应用在不同的源（域、协议或端口）之间进行安全的跨域数据传输。在前后端分离的架构中，前端通常运行在不同的域上，因此需要后端服务支持CORS以允许前端应用访问后端API。如图所示：

```
@Bean
public CorsFilter corsFilter() {
    CorsConfiguration config = new CorsConfiguration();
    config.addAllowedOrigin("*");
    config.setAllowCredentials(true);
    config.addAllowedMethod("*");
    config.addAllowedHeader("*");
    config.addExposedHeader("token");
    UrlBasedCorsConfigurationSource configSource = new UrlBasedCorsConfigurationSource();
    configSource.registerCorsConfiguration( path: "**", config);
    return new CorsFilter(configSource);
}
```

- `config.addAllowedOrigin("*")`：允许来自任何源的请求。"*"表示允许所有域的请求。
- `config.setAllowCredentials(true)`：允许发送cookies。用于需要认证的请求。
- `config.addAllowedMethod("*")`：允许所有HTTP方法的请求，如GET、POST、PUT、DELETE等。
- `config.addAllowedHeader("*")`：允许所有请求头的发送。

- `config.addExposedHeader("token")`：添加了token作为可暴露的响应头。

(5) 常用量的封装：封装了常见的用量，例如错误信息提示msg，状态码code，数据data等。

```
2 usages
public final static String CODE = "code";
/** 返回给页面上的数据 */
1 usage
public final static String DATA = "data";
/** 常量current*/
3 usages
public final static String CURRENT = "current";
/** 默认当前页*/
1 usage
public final static Long DEFAULT_CURRENT = 1L;
/** 常量pageSize*/
3 usages
public final static String PAGE_SIZE = "pageSize";
/** 默认每页数量*/
1 usage
public final static Long DEFAULT_PAGE_SIZE = 10L;
/** 成功状态码*/
1 usage
public final static Integer RESULT_CODE_200 = 200;
/** 后台服务器错误*/
2 usages
public final static Integer RESULT_CODE_500 = 500;
```

(6) 全局异常处理：当任何控制器方法抛出异常时，这个全局异常处理器会被触发，并捕获异常。

`@RestControllerAdvice`

```
public class GlobalExceptionHandler {

    no usages
    @ExceptionHandler(value = Exception.class)
    public ReturnMsg handle(Exception e) {
        e.printStackTrace();
        return ReturnMsg.error(e.getMessage());
    }
}
```

(7) 前后端分离：uniapp前端应用通过HTTP请求与后端API进行通信。

使用uniapp提供的网络请求API（如uni.request）来发送请求到后端API，并接收响应。

八、系统实现

8.1 核心功能实现

8.1.1 用户端小程序功能

8.1.1.1 登陆注册功能

(1) 登录注册功能实现过程

登录注册功能是系统用户认证与授权模块的核心组成部分。为了确保用户身份验证的安全性和高效性，本系统采用基于JWT（JSON Web Token）的身份验证机制。在用户登录成功后，服务器将生成一个JWT令牌，此令牌用于标识用户身份，并在后续请求中通过前端携带进行认证。此外，用户的密码采用BCrypt加密技术进行散列加密后存储在数据库中，确保用户密码信息的高度安全性。

在实现过程中，系统首先对用户输入的用户名和密码进行完整性检查，确保传入参数有效。接着，服务器利用QueryWrapper条件构建器在数据库中查询是否存在与输入信息匹配的用户记录。查询限定为一条结果以提高效率，当查询结果为匹配用户时，服务器将相关信息生成并储存到一个Map对象中，包括用户头像、简介、姓名和角色等细节。随后，系统生成一个唯一的UUID作为JWT令牌，并将其与用户信息一起存储在userInfoMap内用于临时会话管理。最终，服务器将生成的JWT令牌以JSON格式返回给前端，前端则保存该令牌供后续的HTTP请求进行身份认证使用，从而实现用户的安全访问。

(2) 登录注册功能结果展示

登录注册功能为用户与系统交互的起始阶段，用户采用手机号或邮箱简单地注册，系统将采用加密途径对用户密码进行存储，保证数据的安全性，用户输入恰当的账户信息后，凭借简化界面迅速核实身份，成功后可获得基于JWT的会话实时生成的安全令牌，准许其顺畅地访问其他功能模块。该功能设计不仅让用户体验得到提升，并且提

升了系统的安全保障水平，登录注册功能效果如下图所示：



8.1.1.2 健康数据记录及分析

(1) 健康数据记录与分析实现过程

数据记录： 用户通过表单录入身高、体重、脉率等健康数据，提交表单后调用后端接口，将数据存储在数据库对应的健康数据表中。

数据分析： 前端结合第三方图表库（如ECharts）生成健康数据的可视化折线图，展示历史数据变化趋势。

健康数据记录及分析功能包括两个主要模块：记录与分析。用户凭借表单填写自身的健康数据，后台接收数据，接着把这些数据存到数据库的健康数据表里，采用调用相关接口，跟第三方图表库（如ECharts）进行结合，为前端产出健康数据的可视化图表，以呈现用户健康数据的历史变化走向，同时提供健康状况的分析以及个性化建议。

用户提交数据后，系统把数据存进了数据库，数据存储时囊括学生ID等字段数据，以达成记录的有序梳理。当请求开展健康状况分析时，后台根据用户ID到数据库查询健康及学生信息，整合资料后生成问答式描述文段，系统凭借多线程异步开展操作，把分析问题递交给第三方API（如ApiService.query），获取分析结果和健康建议，接着把分析结果写回数据库对应的记录字段内。

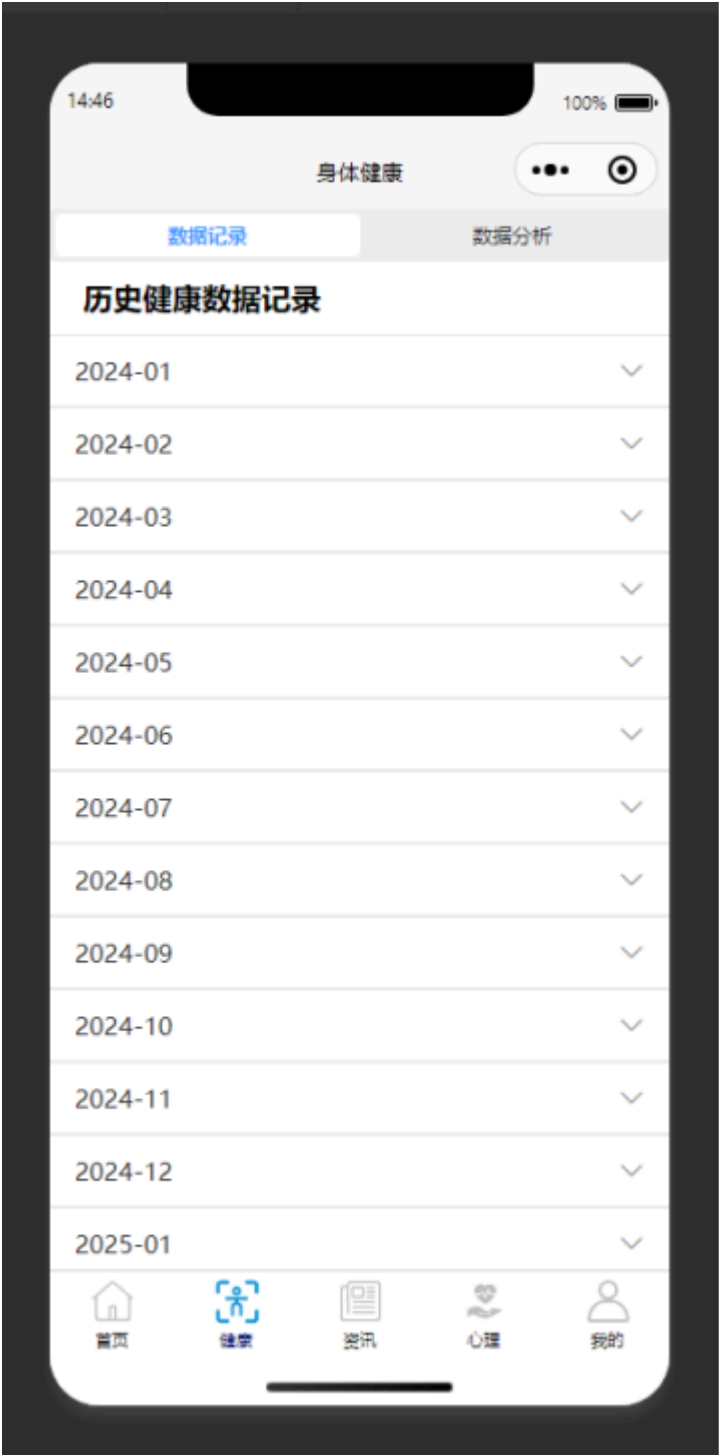
数据的可视化是通过调用特定接口完成的，后端依据学生的ID检索健康相关数据，整理成不同范畴（如身高、体重这类）的时间序列，封装到对象里让前端生成健康折线图。数据处理涉及对各项指标进行解析、分割以及格式化，按照时间点分别把它们存到对应的列表里面去，并跟时间列表实现对应，最终以JSON格式把处理结果返回前端，供图表组件库去生成可视化折线图。

前端采用折线图展示用户健康数据，助力用户直观把握其健康指标在不同时间段的变化状态，同时按照问答分析结果提供个性化的健康小贴士，辅助用户健康管理。通过记录、存储与分析，该功能实现了用户健康数据的追踪和动态呈现，提供了智能化的解决方案。整体实现了数据管理的规范性与可视化呈现的有效性目标，为用户提供了清晰的健康变化参考。

（2）结果展示

健康数据记录模块可让用户方便快捷地记录个人健康信息，用户可输入日常的身高、体重、视力以及脉率之类的健康指标，系统会自动把这些数据存到数据库里，依靠简约的操作界面，用户可简便地完成数据输入。基于这个基础，系统依旧提供数据验证功能，以保障用户录入数据的准确性与完整性。用户定期记录健康数据，可促进形成健康管理习惯，增加自身对健康状况的认知水平，健康数据记录功能效果如下图所

示：



8.1.1.3 心理测试功能

(1) 心理测试功能实现过程

用户可以通过关键词或类别检索心理测试试卷，选择合适的试卷后完成答题。

系统计算用户作答的分数，根据预设规则返回心理健康评估结果。

心理测试功能旨在凭借用户答题后分数的计算以及规则匹配，进行心理健康评估而后生成反馈，同时把关键信息保存下来，接着给对应的心理老师发送心理健康提醒，该

功能的逻辑明白地在核心代码里体现，具体如下：/saveGrade接口接收用户提交的测试记录（RecordsDTO），根据记录中的试卷ID查询相关的心理测试规则

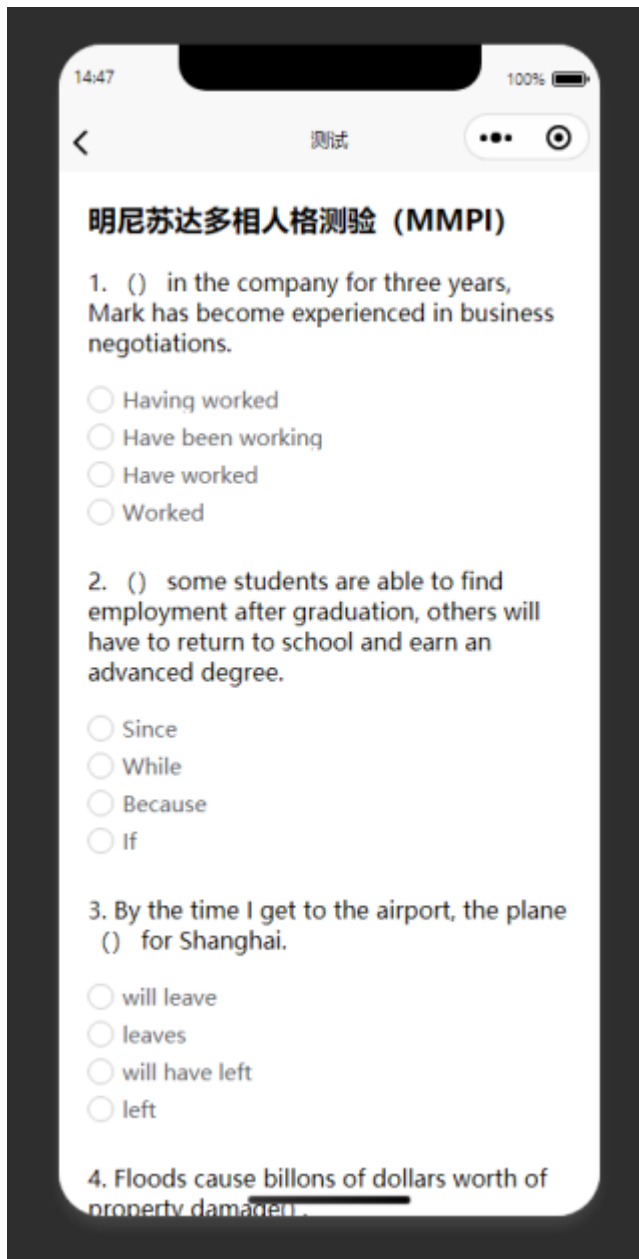
（TestRulesDTO）。查询使用了LambdaQueryWrapper构建条件，系统筛选规则列表，以确保评估逻辑与试卷分数的对应关系。当规则列表为空时，系统仅返回学生得分作为结果；当规则匹配时，系统会根据用户分数所处的区间，提取对应的评估信息（如心理状态或建议）。

若匹配到适用的规则之后，系统动态产出两条反馈信息：第一条是直接反馈给用户的得分评估结果，该结果是把分数与心理测试规则中定义的描述信息相结合得出的；第二条是给心理老师的提醒类系统通知，明确包含学生姓名、答题所得分数，以及基于规则映射得到的心理测试结果评估讲解，代码内的字符串拼接逻辑把这些信息拼凑成完整的通知文本。

生成的评估反馈（包括时间戳、评估内容）会保存到数据库里，以保证后续跟踪及查询的可操作性。此外，为了实现即时提醒功能，代码中包含了多线程实现的邮件发送逻辑，使用mailUtil.sendSimpleMail函数将学生的心理测试评估结果通知到对应的心理老师邮箱，该异步操作让邮件发送的效率得到提高，同时保证主线程不会由于等待网络相关操作而受阻。

（2）结果展示

心理测试模块给出了一种简便手段来评估个人的心理健康情形，用户凭借关键字查找心理测试试卷，系统会展现出契合用户要求的测试清单，用户选定某一试卷之后可开始进行作答，系统会依据预设的判断规则给出评估结果，该功能不仅为用户心理健康提供了评估工具，还带动人们对心理健康的重视与关注，心理测试的组合使得用户能更清楚地知晓自身心理状况，从而得到必需的建议与指导，心理测试功能效果如下图所示：



8.1.1.4 健康资讯推荐

健康资讯推荐模块为用户奉上个性化的健康信息和咨询，用户能借助关键词或是不同的分类查找健康相关资讯，例如饮食建议、运动指导、心理健康等。系统会依据用户的检索行为和兴趣作出智能推荐，从而辅助用户获取最相关的资讯，该功能的设计目的是提升用户的健康素养，推动用户主动去获取专业健康知识，增强其应对健康管理

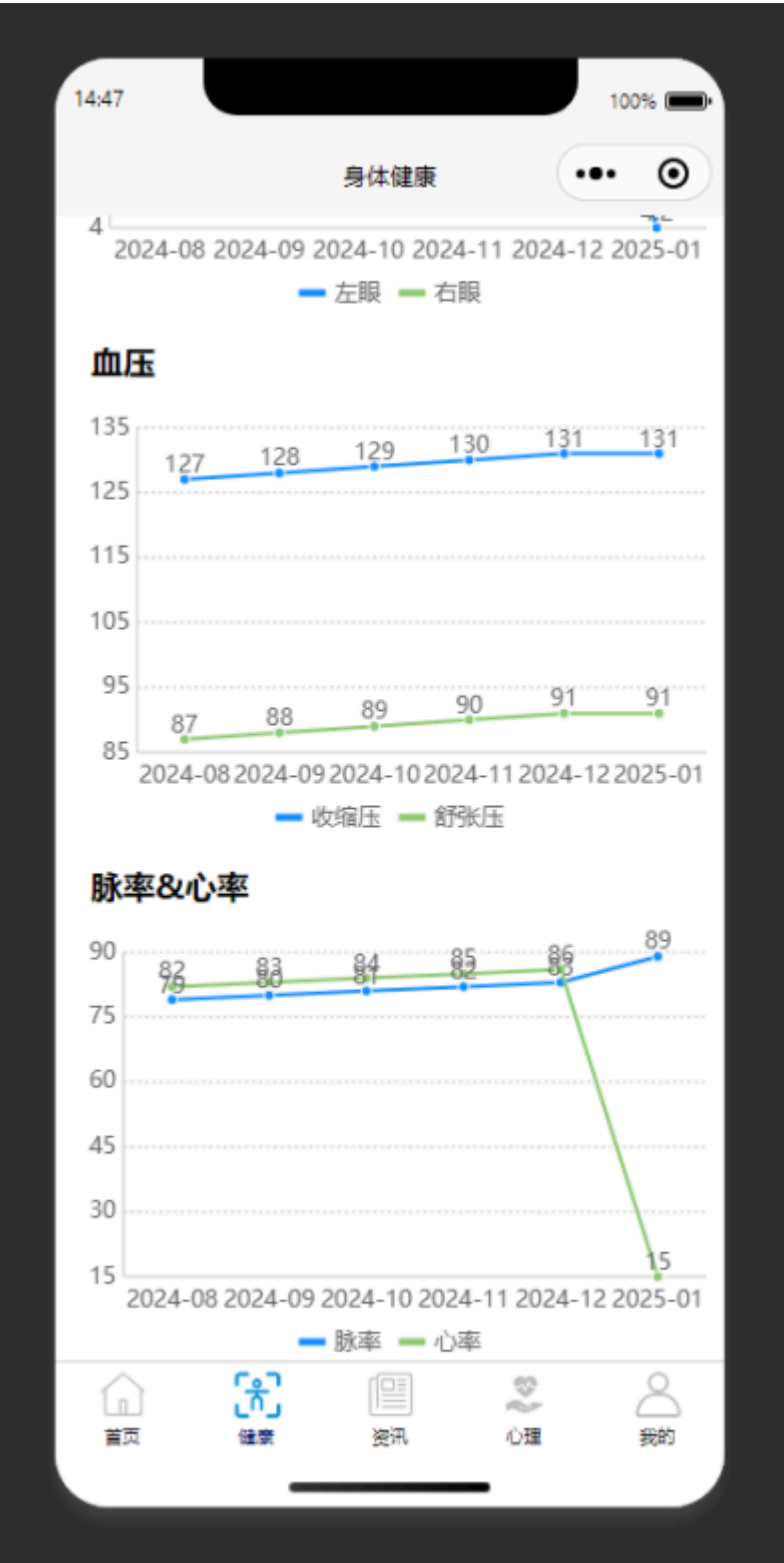
的能力，健康资讯推荐功能效果如下图所示：



8.1.1.5 数据分析

数据分析模块给用户呈现了一览历史健康数据变化的直观画面，系统把用户的身高、体重、视力以及脉率和心率信息用折线图呈现出来，让用户能清楚看到自身健康指标的变化趋势，该功能方便用户对健康变化进行跟踪，还可以在数据出现异常之际，及

时引起用户留意，现代可视化技术与数据分析功能的实现相结合，方便用户找出健康风险并制定管理办法，数据分析功能效果如下图所示：



8.1.2 教师web端功能实现

8.1.2.1 健康管理

(1) 健康管理实现过程

试卷管理：教师可以新建心理测试试卷、编辑题目内容、设置分值权重，并通过权限系统限制试卷的编辑与发布操作。

判断规则管理：教师根据心理健康评估的要求设定分数范围与对应评估结果，并将规则存储到相应的数据库表中。

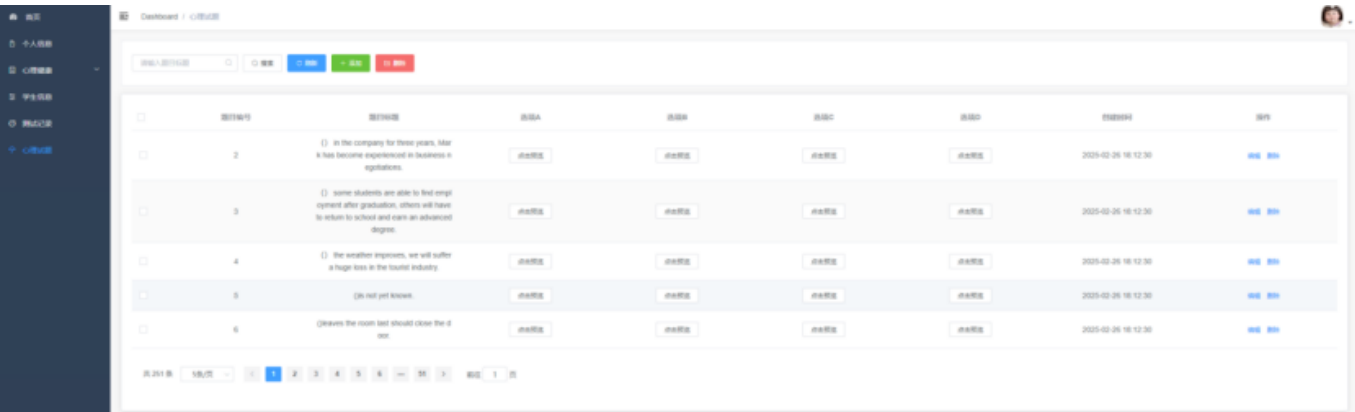
首先是试卷查询功能，通过/all/testPaper/{teaId}接口，系统根据教师ID（teaId）动态查询该教师创建的所有试卷。实现过程使用LambdaQueryWrapper构建查询条件，以筛选出与教师相关的试卷数据。随后，循环遍历查询结果，将每个试卷的唯一标识（ID）和标题（Title）提取并封装为BaseSelectEntity对象，最终以列表形式返回给前端。这种设计不仅支持灵活查询，还便于教师快速定位和管理试卷。

试题查询功能通过/all/questions/{teaId}接口实现，逻辑上类似于试卷查询。系统根据教师ID筛选其创建的所有试题，并将试题的ID与标题构造成结构化字符串（如ID：Title），封装为BaseSelectEntity对象，返回试题列表。试题查询的设计同样依托动态查询条件和数据封装，支持教师快速选择试题并进行后续操作。

两个接口的代码实现均采用了动态查询和结构化数据封装。查询使用LambdaQueryWrapper创建条件，确保高灵活性；结果封装为统一的BaseSelectEntity结构，保障了数据返回的一致性，同时便于前端展示和使用。整体上，这些接口为心理测试试卷和题目管理功能奠定了基础。

(2) 结果展示

这个模块赋予教师对心理测试试题进行添加、修改或删除的权限，教师可根据心理测试的需要打造出多样化的题库，以适应不同学生心理测试的定制化诉求，心理试题管理功能效果如下图：



8.1.2.2 学生信息管理

(1) 学生信息管理实现过程

教师可以根据班级或关键字检索学生信息，并查看学生的健康数据和心理测试记录。

学生信息管理功能通过一个统一的接口实现了增、删、查、改操作，功能逻辑基于 MyBatis-Plus 框架实现，代码设计兼具简洁与灵活性。接口接收传入的学生信息数据实体，首先通过判断是否包含 ID 字段区分操作类型：若存在 ID，则视为修改操作；否则为新增操作。在数据处理前，系统分别调用 beforeSave 或 beforeUpdate 进行预处理以确保操作规范性。核心逻辑依托 MyBatis-Plus 的 saveOrUpdate 方法，自动选择新增或更新操作，根据数据状态统一处理，无需手动编写SQL语句。操作完成后，通过调用后置处理方法（afterSave 或 afterUpdate）执行额外的业务逻辑，如数据初始化或状态调整，进一步加强功能的扩展性和可定制性。最终，统一返回操作结果消息，无论成功或失败，都给予清晰的反馈。整个学生信息管理功能设计高度模块化、灵活化，不仅简化了教师的数据操作流程，还为后续扩展业务逻辑提供了可靠的技术支持。

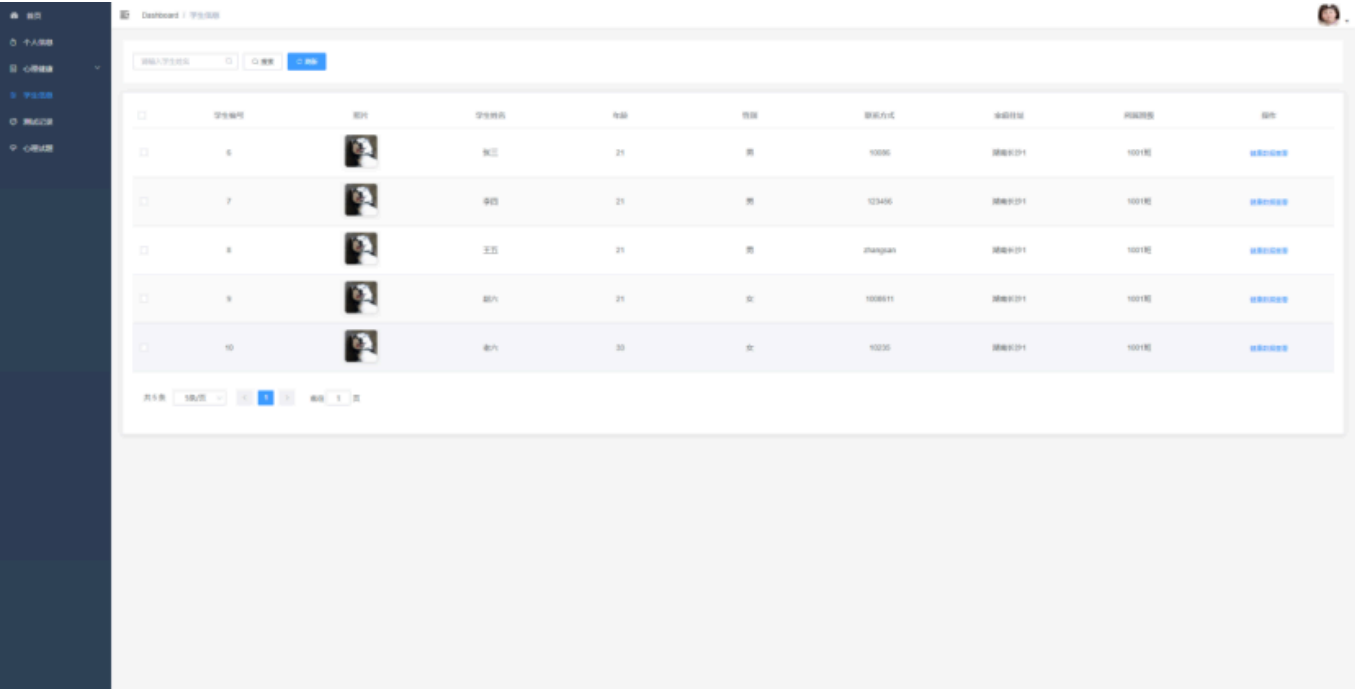
(2) 结果展示

学生管理模块属于校园智能健康管理平台的核心功能范畴，该模块配置了全面的学生信息管理接口，目的为对学生的基本信息和健康数据开展系统的维护与分析，管理员可借助此模块查看学生的个人资料，像姓名、年龄、性别、班级这类基础信息，确保信息的准确性和完备性。

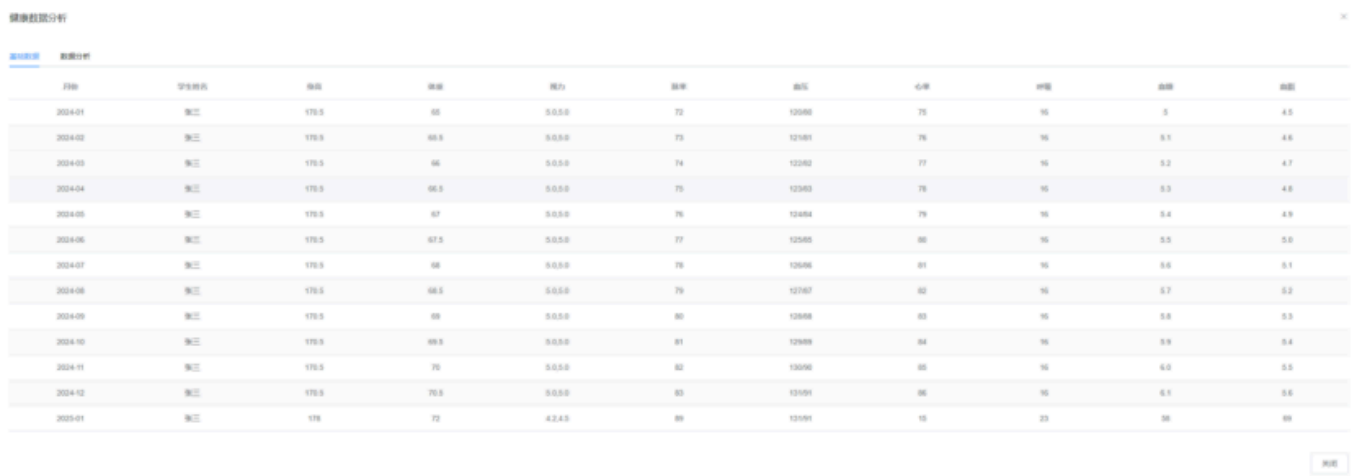
除了实施基本信息管理，学生管理模块还把重点放在学生健康数据的记录和跟踪上，各个学生的健康数据以月份为单位作系统展示，管理员可查看学生各个月份里的基础健康指标，如身高、体重、心率、视力等。这些数据不仅以列表形式进行展示，便于快速浏览；同时又借助折线图动态地展示，以此助力管理员迅速捕捉数据的变化趋势与异常问题。

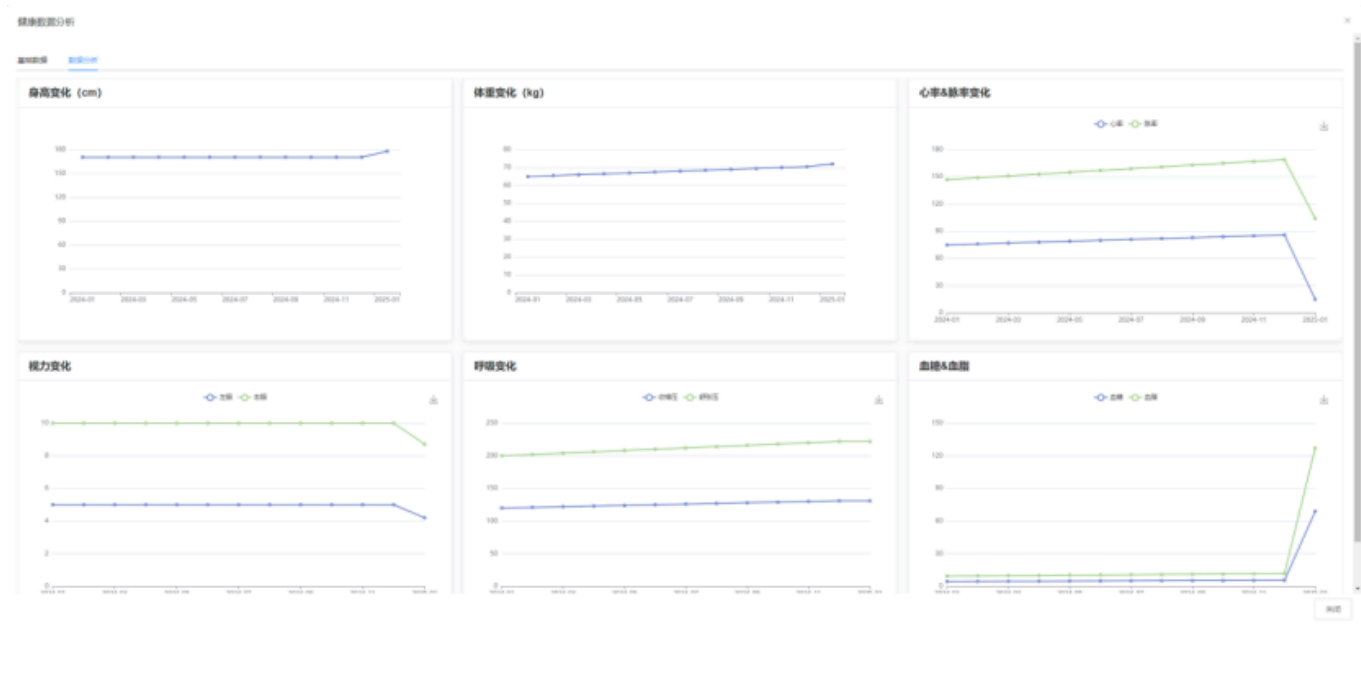
这个模块目的是协助学校管理层精准掌握学生群体的健康状态，从而在有必要的时候采取预防手段，所积累的健康数据可作为制定健康教育策略及营养方案的参考凭据，依靠整合的数据分析及报告功能，学生管理模块极大提升了校园健康管理的科学性，使个体健康状况可以一直得到关注与改善，学生信息管理功能效果如下：

学生信息管理功能效果图：



健康数据分析-数据分析功能效果图：





8.1.2.3 测试记录管理

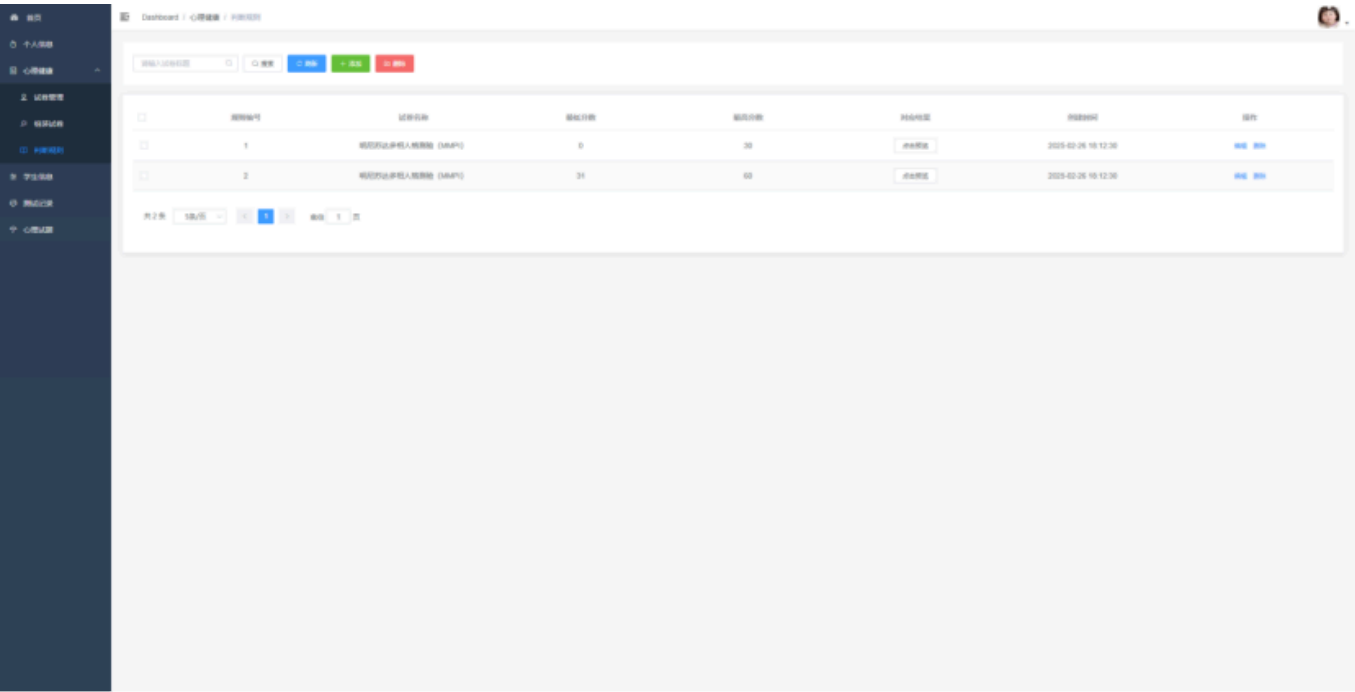
该功能可向教师提供所有学生完成心理测试的详细记录，涉及测试的时间、试卷模样以及测试反馈，教师可迅速查看历史记录，方便对学生心理状况做长期跟踪与趋势解析，测试记录管理功能效果如下图所示：

ID	学生姓名	学生照片	试卷标题	试卷类型	总分	考试次数	测试时间	操作
10	张三		积极心理量表 (SAPS)		30	10	2025-02-26 15:12:30	详情
11	张三		卡特尔16种人格因素量表 (16PF)		10	0	2025-02-26 15:12:30	详情
12	张三		积极心理量表 (SAPS)		30	0	2025-02-26 15:12:30	详情
13	张三		积极心理量表 (SAPS)		30		2025-02-26 15:12:30	详情
14	张三		积极心理量表 (SAPS)		30	20	2025-02-26 15:12:30	详情

8.1.2.4 判断规则

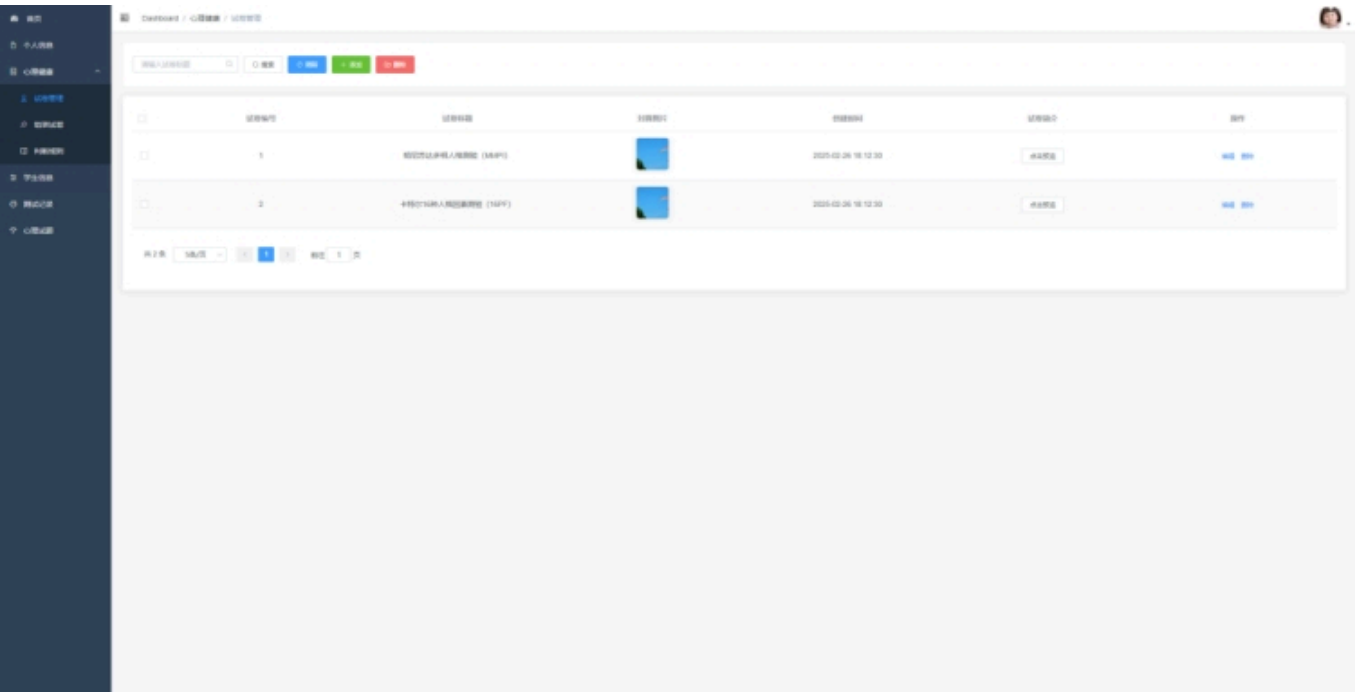
教师端有判断规则相关功能，支持教师制定心理测试结果的评定标准，教师能根据试卷与健康数据去设置归类规则，做到对测试结果的精准剖析，为后续的心理干预给予

科学凭据，判断规则功能效果如下图所示：



8.1.2.5 试卷管理

此功能用于管理心理测试试卷，教师创建新的心理测试试卷，并对试卷内容进行统一管理。试卷管理功能效果如下图所示：



8.1.3 管理Web端功能实现

以知识分类管理和健康数据管理这两个功能的实现为例。

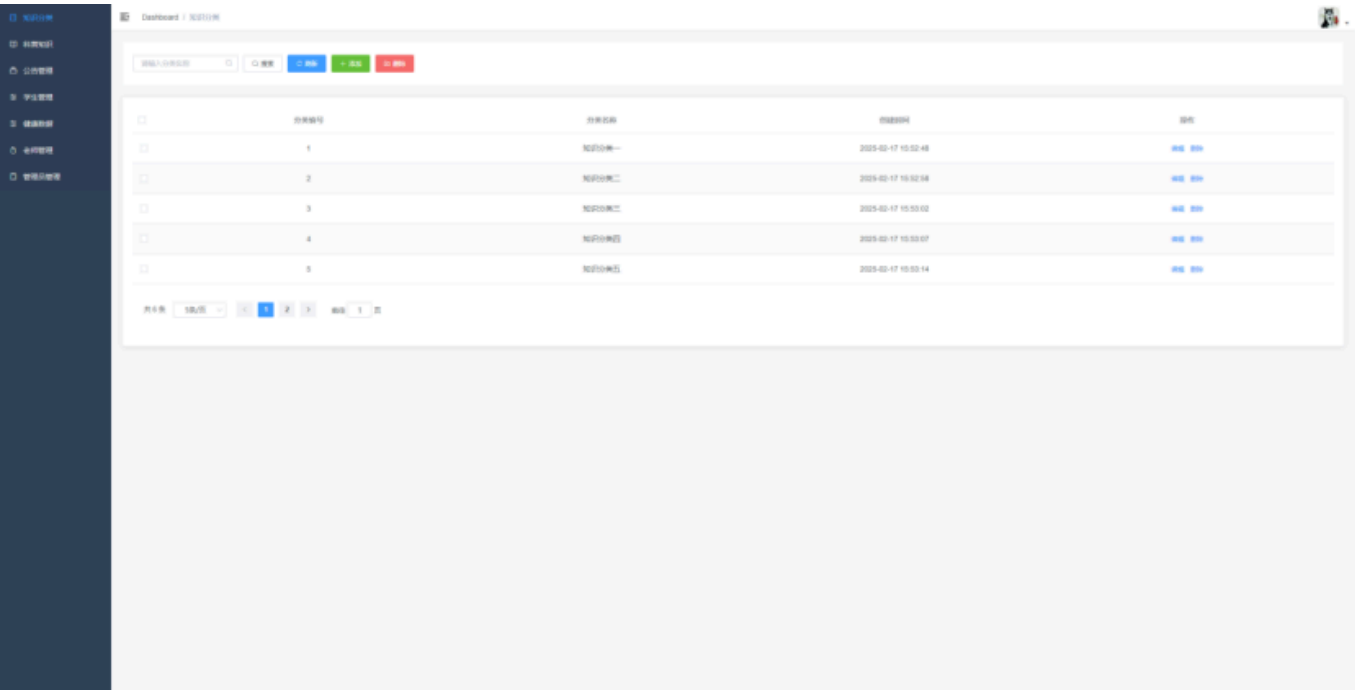
8.1.3.1 知识分类管理

(1) 功能实现过程

健康知识分类管理功能通过结构化的增删查改接口实现，提供了简洁且高效的支持。在实现过程中，管理员新增或编辑知识分类时，通过调用baseSaveOrUpdate接口，系统会根据传入实体对象的ID字段决定是新增或更新。beforeSave和beforeUpdate方法保证数据的完整性和规范性，而MyBatis-Plus的saveOrUpdate简化了操作逻辑。处理完成后，通过afterSave或afterUpdate进行后续逻辑处理，例如更新缓存或通知相关模块状态变化。删除操作通过baseRemoveById接口进行，确保在调用beforeRemoveById进行数据安全性检查后，再使用MyBatis-Plus的removeById执行实际删除，最终返回操作结果以便进一步处理。分类分页查询通过pageByWrapper接口实现，使用QueryWrapper构造闲置查询条件，并基于页面参数高效执行分页查询，结果支持后置处理以增强展示效果。

(2) 结果展示

健康知识分类管理模块实现了管理员流畅地新增、编辑和删除健康知识分类的能力。通过标准化接口，保障了操作的安全和数据的一致性。分页查询与动态条件结合提升了健康信息的检索效率和分类展示效果，使得用户能够快速定位和管理相关知识类别。模块化的设计增强了功能的可扩展性，确保管理员在管理健康知识分类时更高效、更安全，并能够快速适应业务需求的变化。知识分类管理功能效果如下图所示：



8.1.3.2 健康数据管理

(1) 功能实现过程

健康数据管理模块通过多接口和方法的协同工作，实现了学生健康数据的收集、处理、可视化展示以及智能化分析。其中，在健康数据的可视化分析功能中，系统通过@PathVariable注解获取传入的学生ID作为查询条件，并利用该ID创建HealthyDTO对象，通过调用baseService.findList()方法从数据库中检索所有相关记录。然后，系统初始化各种指标集合，例如月份列表、身高列表等，进入数据解析逻辑后，会遍历查询结果，将每个健康数据点解析到各自的指标集合中。例如，对于视力数据，系统会通过split方法拆分字符串，提取出左右眼视力值。类似地，血压数据会通过斜杠分割获得收缩压和舒张压数值。解析完成后，所有指标集合被赋值给EchartsVO对象，最终接口通过ReturnMsg.ok()将封装后的对象响应给前端，用于生成动态的健康数据图表。

AI健康评估功能借助一个独立的接口实现了实时数据分析和智能化健康报告生成。该接口首先根据学生健康记录的ID从数据库中查询指定记录，并通过String.format()方法动态构建问题模板，由问题字符串表达学生的基本信息以及健康记录关键数据。接着，系统启动一个新线程，异步调用ApiService.query()方法向外部AI分析服务发送生成的问题字符串，以获取分析结果。结果被赋值到健康记录的content字段，并通过baseService.updateById()方法更新到数据库。这种异步处理方式既保证了系统高效运行，也避免了因数据分析造成页面响应的阻塞。

为了便捷地录入大量健康数据，模块还支持批量上传功能。管理员可以通过Excel文件上传健康数据，系统借助EasyExcel库解析文件内容。文件解析过程中，每一行数据由自定义的ExcelListener类处理，该类实现了AnalysisEventListener接口，在invoke方法中将Excel数据映射到HealthyDTO对象，并对特殊字段（如血压）进行自动格式化转换，将收缩压和舒张压拼接为统一格式的字符串。最后，处理后的数据通过healthyService.save()方法保存至数据库，实现了批量录入的高效性。

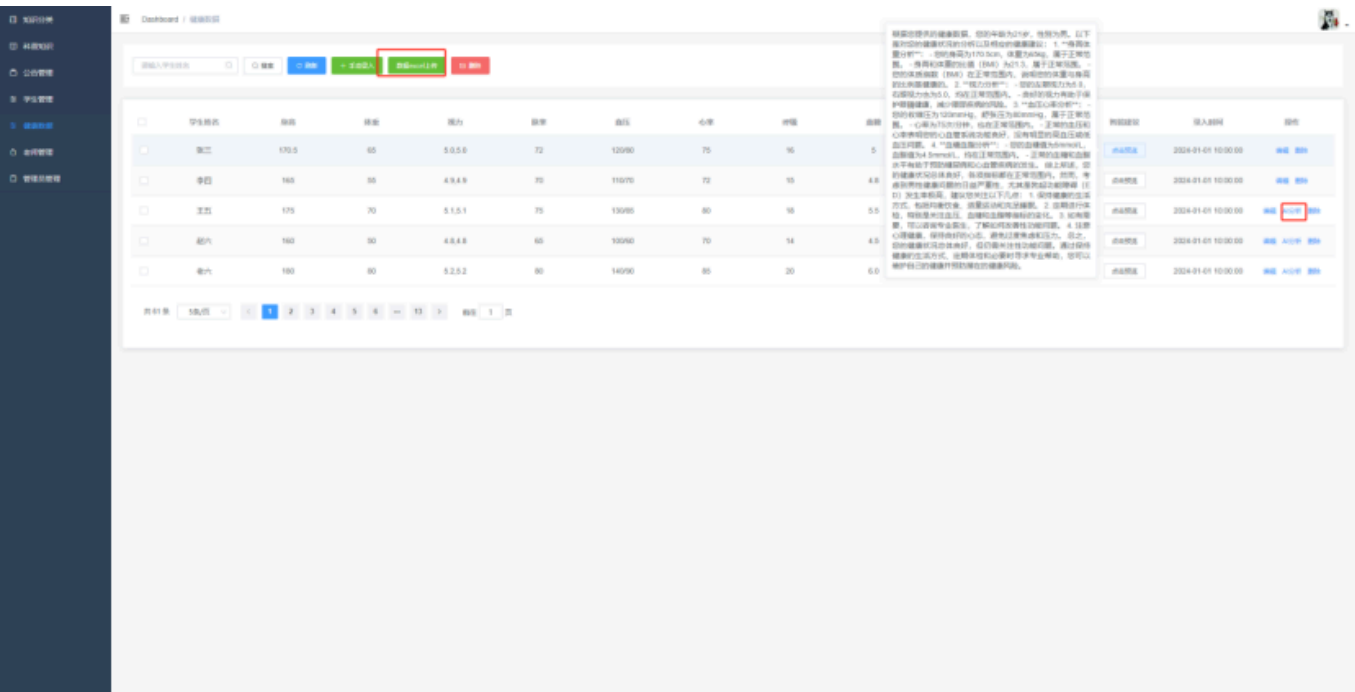
（2）结果展示

健康数据管理模块承担收集、综合与智能化分析学生健康数据的工作，给出了一整套高效的数据处理及可视化工具，目的是保障数据的完整度和实用功效，管理员可借助此模块开展健康记录的维护工作，如信息录入、编辑、删除等操作，保障所有数据可以及时更新并反馈在系统里面。

特别要说明的是，健康数据管理模块支持Excel批量数据上传功能，准许管理员在较短时间内导入大批健康数据，大幅增进数据处理的效率。同时，该模块创新性地引入了AI数据分析工具，凭借人工智能技术对存储的健康数据做深入剖析，AI分析工具凭借健康数据生成健康情况评估报告，进而给出针对个人的健康建议，使处理后的数据具有更高的实用及指导意义。

健康数据管理模块不光提升了数据管理的效率跟精准度，还扩充了数据的应用范畴，为学校健康教学与实践给予了强有力的支撑，该模块呈现出的智能化特质，确保了校

园健康管理在信息化时代背景下的先进性与前瞻性。依靠健康数据的及时解析与有效应用，学校不光能对健康管理流程加以优化，还能给学生全生命周期的健康管理打下坚实根基，健康数据管理功能效果如下图所示：



8.2 代码结构

后端代码目录结构：

```
src
├── main
│   ├── java
│   │   └── com.example.healthy
│   │       ├── controller    # 控制层：API接口
│   │       ├── service      # 业务层：核心功能逻辑处理
│   │       ├── mapper        # 数据访问层：MyBatisPlus接口映射
│   │       ├── entity        # 实体类：数据库表的实体映射
│   │       ├── config        # 配置文件：跨域、JWT、日志等配置
│   │       └── exception     # 全局异常处理
│   └── resources
│       ├── application.yml  # 项目配置文件：数据库连接、端口号等
│       └── mapper.xml       # MyBatisPlus自定义SQL映射
```

前端代码目录结构：

```
src
├── assets      # 静态资源文件，如图片、样式
├── components # 自定义组件，如数据表、表单
├── pages       # 页面：每个模块对应一个文件夹
│   ├── home   # 首页模块
│   ├── health # 健康管理模块
│   └── personal # 我的模块（个人中心）
├── store       # Vuex状态管理
├── router      # 前端路由配置
├── utils       # 工具类：日期格式化、HTTP请求封装等
├── App.vue     # 全局根组件
└── main.js     # 程序入口文件
```

九、系统测试

9.1 测试策略与计划

系统测试的主要目的是让校园智能健康管理平台在不同环境及使用条件下，其功能性、性能和稳定性达到预期要求和用户需求，旨在通过系统化的测试方法发现系统潜在的问题与缺陷，并验证已修复缺陷的有效性，以此提升系统可靠性以及用户体验。

功能测试是系统测试过程中的一个关键环节，旨在验证校园智能健康管理平台实现的每项功能是否符合设计规格和用户需求。通过功能测试能够确保各模块和组件在实际使用中的准确性与完整性。

9.2 单元测试与集成测试

测试表：

用例编号	测试用例名称	测试目标	前置条件	测试步骤	测试结果
TC-001	用户注册功能测试	确认用户可以成功注册并创建账户	系统正常运行，用户未注册	1. 打开注册页面 2. 输入有效的用户名、密码、邮箱 3. 提交注册	用户成功注册并收到确认邮件
TC-002	登录功能测试	验证注册用户可通过正确的	用户已注册且账户已激	1. 打开登录页面 2. 输入正确的用	成功登录，进入用户主

用例编号	测试用例名称	测试目标	前置条件	测试步骤	测试结果
		用户名和密码访问系统	活	户名和密码 3. 点击登录按钮	页
TC-003	记录健康数据测试	确认用户能够记录并保存健康数据	用户已登录进入系统首页	1. 选择健康数据记录选项 2. 输入相关数据 3. 点击保存按钮	健康数据被成功保存, 显示确认信息
TC-004	数据分析功能测试	验证用户可查看其健康数据的分析结果	用户已登录并有健康数据记录	1. 进入数据分析页面 2. 查看图表及分析结果	正确显示健康数据分析图表
TC-005	心理测试功能测试	确认心理测试功能能够正常运行并输出结果	用户已登录进入系统首页	1. 进入心理测试页面 2. 完成所有心理测试问题 3. 提交测试	系统反馈心理评估结果
TC-006	管理员查看用户信息测试	验证管理员可以查看用户健康和心理信息	管理员已通过身份验证进入管理端	1. 进入用户信息管理页面 2. 查询指定用户信息	正确显示用户的详细健康记录
TC-007	系统安全性测试	测试系统能抵御常见的安全攻击 (如SQL注入)	环境准备充分, 系统处于可攻击的状态	1. 在输入框尝试SQL注入 2. 提交恶意输入	系统未被攻击, 安全性保护有效
TC-008	用户反馈功能测试	确认用户可以成功提交反馈并获得响应	用户已登录进入系统首页	1. 进入反馈页面 2. 输入反馈内容 3. 提交反馈	系统反馈接收成功并提示处理结果

经过全面系统的测试，包括功能、性能和安全性测试，校园智能健康管理平台的各项功能均达到了设计要求。测试发现的少数问题已及时修复并通过复测验证，系统性能在高负载环境下表现优秀，安全防护有效。该平台符合当前部署标准，具备上线条件。

十、功能展示

详细的功能展示与描述见本文第八章：系统实现。本章仅附系统功能展示图。

10.1 移动端

14:52

100% 



注册



您好，
欢迎使用健康管理小程序

手机号

请输入您的手机号

密码

请输入您的密码

确认密码

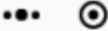
请再次输入您的密码

提交

14:46

100%

身体健康



数据记录

数据分析

历史健康数据记录

2024-01	▼
2024-02	▼
2024-03	▼
2024-04	▼
2024-05	▼
2024-06	▼
2024-07	▼
2024-08	▼
2024-09	▼
2024-10	▼
2024-11	▼
2024-12	▼
2025-01	▼



首页



健康



报告



心理



我的

14:47

100%



测试



明尼苏达多相人格测验 (MMPI)

1. () in the company for three years, Mark has become experienced in business negotiations.

- ☐ Having worked
- ☐ Have been working
- ☐ Have worked
- ☐ Worked

2. () some students are able to find employment after graduation, others will have to return to school and earn an advanced degree.

- ☐ Since
- ☐ While
- ☐ Because
- ☐ If

3. By the time I get to the airport, the plane () for Shanghai.

- ☐ will leave
- ☐ leaves
- ☐ will have left
- ☐ left

4. Floods cause billions of dollars worth of property damaged.

14:46

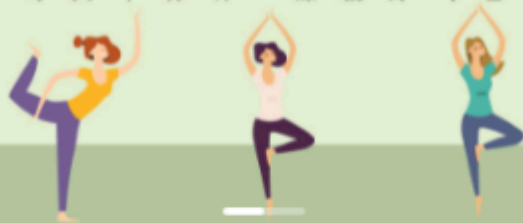
100%

首页



瑜伽 养 生 季

每天十分钟 练就 好 气 色



健康资讯

查看更多 >



如何拥有
健康的生活方式

预防诺如病毒感染，这种“隐秘”传播需留意

2023-10-09 14:00:00



如何拥有
健康的生活方式

开学季校园传染病防控“十要”

2023-10-08 12:00:00



如何拥有
健康的生活方式

春节期间如何预防病从“口”入？

2023-10-07 10:00:00



如何拥有
健康的生活方式

为何冬季容易出现情绪和睡眠障碍？应该怎么办？

2023-10-06 20:00:00



首页



健康



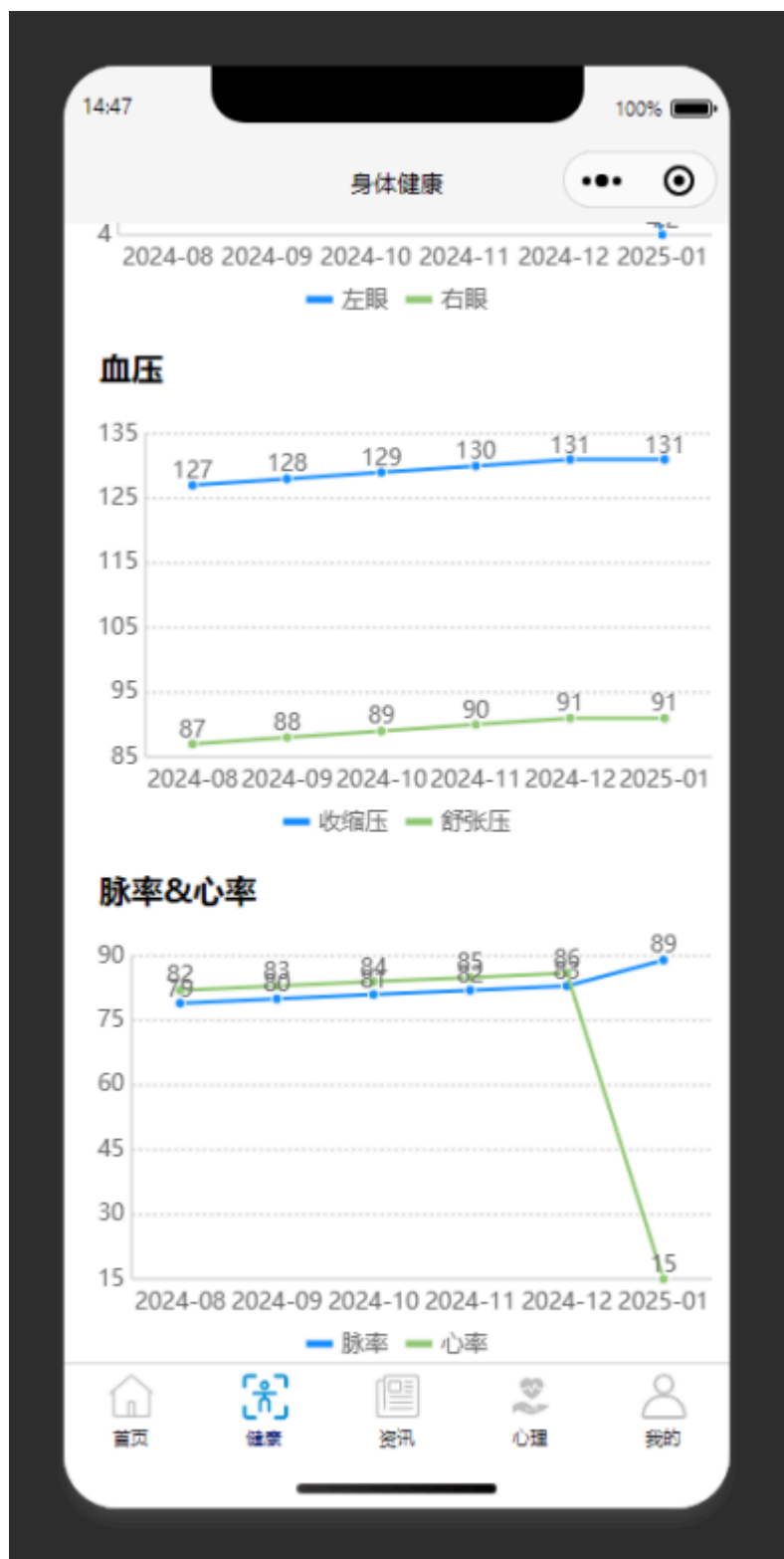
资讯



心理



我的



10.2 PC端

首页

个人中心

心理档案

学生档案

测试记录

心理记录

Dashboard / 心理记录

请输入测试记录

搜索

全部

心理

10 条

<input type="checkbox"/>	题目编号	题目内容	选项A	选项B	选项C	选项D	创建时间	操作
<input type="checkbox"/>	2	() In the company for three years, Mr. X has become experienced in business negotiations.	查看详情	查看详情	查看详情	查看详情	2025-02-26 16:12:30	删除 详情
<input type="checkbox"/>	3	() Some students are able to find employment after graduation, others will have to return to school and earn an advanced degree.	查看详情	查看详情	查看详情	查看详情	2025-02-26 16:12:30	删除 详情
<input type="checkbox"/>	4	() The weather improves, we will suffer a huge loss in the tourist industry.	查看详情	查看详情	查看详情	查看详情	2025-02-26 16:12:30	删除 详情
<input checked="" type="checkbox"/>	5	(do not yet know)	查看详情	查看详情	查看详情	查看详情	2025-02-26 16:12:30	删除 详情
<input type="checkbox"/>	6	(Heaven the room test should clear the d	查看详情	查看详情	查看详情	查看详情	2025-02-26 16:12:30	删除 详情

共 10 条

1 2 3 4 5 6 7 8 9 10

 第 1 页

首页

个人中心

心理档案

学生档案

测试记录

心理记录

Dashboard / 学生档案

请输入学生档案

搜索

全部

<input type="checkbox"/>	学生编号	照片	学生姓名	年龄	性别	联系方式	身份证号	所属院系	操作
<input type="checkbox"/>	6		张三	21	男	10090	10090123456789	1001班	删除并同步
<input type="checkbox"/>	7		李四	21	男	123456	1234567890123456	1001班	删除并同步
<input type="checkbox"/>	8		王五	21	男	zhangsan	1234567890123456	1001班	删除并同步
<input type="checkbox"/>	9		赵六	21	女	1008811	1008811123456789	1001班	删除并同步
<input checked="" type="checkbox"/>	10		赵六	33	女	10230	10230123456789	1001班	删除并同步

共 5 条

1 2 3 4 5

 第 1 页

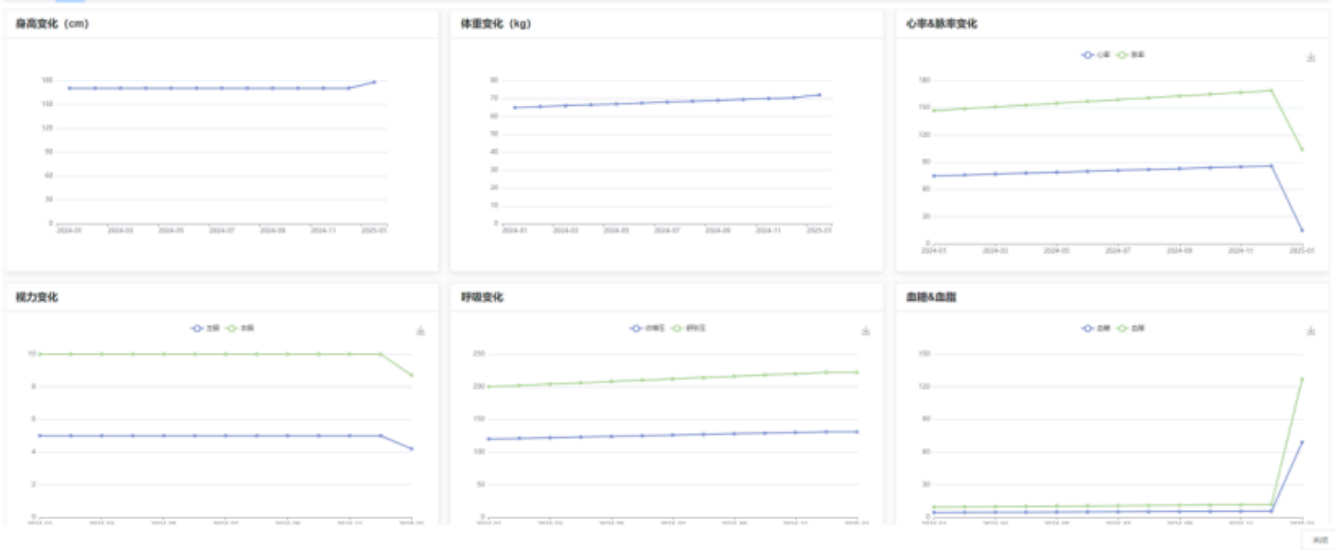
健康数据监测

数据详情

日期	学生姓名	身高	体重	视力	心率	血压	心率	呼吸	血糖	血脂
2024-01	张三	175.5	65	5.0/5.0	72	120/80	75	16	5	4.5
2024-02	张三	175.5	66.5	5.0/5.0	73	121/81	76	16	5.1	4.6
2024-03	张三	175.5	66	5.0/5.0	74	122/82	77	16	5.2	4.7
2024-04	张三	175.5	66.5	5.0/5.0	75	123/83	78	16	5.3	4.8
2024-05	张三	175.5	67	5.0/5.0	76	124/84	79	16	5.4	4.9
2024-06	张三	175.5	67.5	5.0/5.0	77	125/85	80	16	5.5	5.0
2024-07	张三	175.5	68	5.0/5.0	78	126/86	81	16	5.6	5.1
2024-08	张三	175.5	68.5	5.0/5.0	79	127/87	82	16	5.7	5.2
2024-09	张三	175.5	69	5.0/5.0	80	128/88	83	16	5.8	5.3
2024-10	张三	175.5	69.5	5.0/5.0	81	129/89	84	16	5.9	5.4
2024-11	张三	175.5	70	5.0/5.0	82	130/90	85	16	6.0	5.5
2024-12	张三	175.5	70.5	5.0/5.0	83	131/91	86	16	6.1	5.6
2025-01	张三	176	72	4.2/4.5	85	135/95	15	23	36	85

加载

基础数据 数据对比



- 首页
- 个人中心
- 健康监测
- 学生信息
- 测试记录
- 测试结果

Dashboard / 测试记录

请输入测试日期

请输入学生姓名

搜索

刷新

<input type="checkbox"/>	编号	学生姓名	学生照片	测试科目	测试日期	总分	考试分数	测试时间	操作
<input type="checkbox"/>	10	张三		科目名称: 多项选择题 (SAP1)		30	10	2025-02-26 18:12:30	查看
<input type="checkbox"/>	11	张三		科目名称: 多项选择题 (SAP2)		10	0	2025-02-26 18:12:30	查看
<input type="checkbox"/>	12	张三		科目名称: 多项选择题 (SAP1)		30	0	2025-02-26 18:12:30	查看
<input type="checkbox"/>	13	张三		科目名称: 多项选择题 (SAP1)		30		2025-02-26 18:12:30	查看
<input type="checkbox"/>	14	张三		科目名称: 多项选择题 (SAP1)		30	20	2025-02-26 18:12:30	查看

共 5 条

1 2 3 4 5

 第 1 页

首页

个人中心

心理健康

心理管理

心理记录

心理测试

心理知识

学生活动

通知公告

心理知识

Dashboard / 心理健康 / 心理管理

请输入关键词

搜索

新增

删除

刷新

	序号	姓名	性别	年龄	出生日期	操作
<input type="checkbox"/>	1	张三	男	20	2020-02-26 18:12:30	编辑 删除
<input type="checkbox"/>	2	李四	女	22	2020-02-26 18:12:30	编辑 删除

共 2 条

1/2

 每页 1 条

首页

个人中心

心理健康

心理管理

心理记录

心理测试

心理知识

学生活动

通知公告

心理知识

Dashboard / 心理健康 / 心理管理

请输入关键词

搜索

新增

删除

刷新

	序号	姓名	性别	年龄	出生日期	操作
<input type="checkbox"/>	1	张三	男	20	2020-02-26 18:12:30	编辑 删除
<input type="checkbox"/>	2	李四	女	22	2020-02-26 18:12:30	编辑 删除

共 2 条

1/2

 每页 1 条

- main
 - java
 - com.example.healthy
 - controller: 控制层: API接口
 - service: 业务层: 核心功能逻辑处理
 - mapper: 数据访问层: MyBatisPlus接口映射
 - entity: 实体类: 数据库表的实体映射
 - config: 配置文件: 跨域、JWT、日志等配置
 - exception: 全局异常处理
 - resources
 - application.yml: 项目配置文件: 数据库连接、端口号等
 - mapper.xml: MyBatisPlus自定义SQL映射

前端代码:

`/psychology_client` 目录下包含前端代码, 如 `package.json` 中定义了前端项目的依赖和脚本, `src/permission.js` 负责路由权限管理。 `/psychology_client_tea` 目录下同样是前端代码, 其结构和依赖与 `psychology_client` 类似。 `/psychology_smart` 目录下也有前端相关代码, 例如 `api` 目录下的 `index.js` 和 `login/loginApi.js` 负责API请求封装, `uni_modules` 包含一些前端组件的配置信息。

- src
 - assets: 静态资源文件, 如图片、样式
 - components: 自定义组件, 如数据表、表单
 - pages: 页面, 每个模块对应一个文件夹
 - home: 首页模块
 - health: 健康管理模块
 - personal: 个人中心
 - store: Vuex状态管理
 - router: 前端路由配置
 - utils: 工具类, 如日期格式化、HTTP请求封装等
 - App.vue: 全局根组件
 - main.js: 程序入口文件

十二、总结与展望

12.1 项目总结

校园智能健康管理平台设计与开发项目顺利达成了将健康管理及心理健康评估融入校园生活的目标。通过利用前端技术（Vue、ElementUI、Uni-app）和后端框架

（Java、SpringBoot、MyBatisPlus），系统构建了用户端小程序、PC教师端和PC管理端三位一体的服务平台。用户可容易地记录健康数据、查看数据分析的最终结果、获取健康相关的资讯，教师与管理人员可高效地对学生的信息和心理健康数据加以管理，该平台的推出，校内师生提供了全面且适用的健康管理方案，带动校园健康资源达到最大化利用效果[18]。

尽管项目已取得显著成效，但在开发和运营中仍然存在一些不足之处。首先必须持续关注系统的性能优化，尤其是当面临高并发访问量时，也许会出现响应迟缓的现象，以后可借助优化数据库查询以及改进缓存机制，提升系统的运行效率。在用户体验设计方面还有进一步提升的空间，应在信息检索便捷性及用户反馈响应速度上进行优化，随着数据安全与隐私保护的重要性愈发凸显出来，需再进一步增强系统的安全防护手段，保证用户数据的机密情况与安全水平。

12.2 技术收获

团队成员在项目开发过程中接触到了多种技术和工具，技术能力得到了显著提升。前端开发人员对 Vue.js 框架的使用更加熟练，掌握了组件化开发、路由管理等高级特性。同时，也学习了一些前端的性能优化技巧，如代码压缩、图片优化等，提高了页面的加载速度。后端开发人员对 Java 语言和 Spring Boot 框架有了更深入的理解，掌握了数据库设计、接口开发、事务处理等技能。此外，还学习了如何使用一些工具进行代码的调试和测试，提高了代码的质量和稳定性。

在项目开发过程中，团队成员的沟通协作能力得到了锻炼。前端和后端开发人员需要密切配合，确保接口的一致性和数据的准确性。同时，团队成员还需要与测试人员、产品经理等进行沟通，及时解决项目中出现的问题。通过这些实践，团队成员的沟通能力和团队协作能力得到了提升。此外，项目的复杂度也促使团队成员不断学习和解决问题，提高了他们的独立思考能力和问题解决能力。

在项目开发过程中，团队积累了丰富的经验。例如，在项目的前期规划阶段，要充分考虑到系统的扩展性和可维护性，合理设计数据库结构和接口。在开发过程中，要注重代码的规范和注释，方便后续的维护和扩展。同时，要建立有效的测试机制，及时发现和解决代码中的问题。此外，还要加强团队成员之间的沟通和协作，提高项目的开发效率和质量。

12.3 问题与反思

在项目开发过程中，遇到了一些问题。前端方面，由于不同浏览器和设备的兼容性问题，导致部分页面在某些浏览器或设备上显示不正常。后端方面，数据库的性能问题是一个比较突出的问题，随着数据量的增加，查询和写入操作的响应时间变长。此外，在项目的集成测试阶段，发现前端和后端之间的接口存在一些数据格式不一致的问题，导致部分功能无法正常使用。

针对前端的兼容性问题，团队采用了多种解决方案。首先，在开发过程中使用了一些前端框架和工具，如 Vue.js 和 PostCSS，来处理浏览器的兼容性问题。其次，对不同浏览器和设备进行了全面的测试，及时发现和解决了一些兼容性问题。针对后端的数据库性能问题，团队对数据库进行了优化，如创建索引、优化查询语句等。同时，也考虑了使用缓存技术来提高数据库的访问速度。对于前端和后端之间的接口问题，团队加强了沟通和协作，明确了接口的规范和数据格式，确保了接口的一致性。

通过这次项目，团队吸取了一些经验教训。在项目开发过程中，要充分考虑到各种可能出现的问题，提前做好应对措施。例如，在项目的前期规划阶段，要对系统的性能进行评估，合理设计数据库结构和接口。在开发过程中，要加强测试，及时发现和解决问题。同时，要建立有效的沟通机制，确保团队成员之间的信息畅通。此外，还要注重代码的规范和注释，方便后续的维护和扩展。

12.4 未来展望

从市场角度来看，伴随社会对健康重视度的上升与政策的推动，校园健康管理已成为学校各级领导关注的重点。未来，平台可拓展至更多的学校及教育机构，甚至涉及社区健康管理范畴，给予全方位的健康数据服务以及心理辅导帮助。并且通过和专业健康机构、心理健康咨询机构达成合作，能够为平台的服务内容增添丰富色彩，开辟更大的市场空间。

在技术层面，校园智能健康管理平台有望凭借前沿技术的引入达成更多创新功能，采用人工智能技术，平台可实现更具智能化的健康数据剖析和个性化健康指引，为用户提供更有针对性的健康管理方案。并且大数据技术的采用将推动平台实现对健康数据的深度挖掘及精准分析，以支撑用户做出有效决策。

12.5 商业化前景

本校园智能健康管理平台具有较高的商业价值。首先，它可以为学校提供全面的健康管理解决方案，提高学校的管理效率和服务质量，从而吸引更多的学生和家长。其次，它可以为健康管理机构和企业提供数据支持和分析服务，帮助他们更好地了解市场需求和用户行为。此外，它还可以通过广告、付费会员等方式实现盈利。

随着人们对健康的关注度不断提高，校园健康管理市场具有广阔的发展前景。学校和家长越来越重视学生的健康问题，希望通过智能化的手段来管理学生的健康。同时，健康管理机构和企业也需要更多的数据和分析服务来支持他们的业务发展。因此，本校园智能健康管理平台具有很大的市场需求。

可能的商业模式包括以下几种。一是向学校和教育机构收取软件使用费用，根据学校的规模和需求提供不同的套餐。二是与健康管理机构和企业合作，为他们提供数据支持和分析服务，收取服务费用。三是通过广告投放来实现盈利，在系统中展示相关的健康产品和服务广告。四是推出付费会员服务，为会员提供更加个性化的健康管理服务和优惠活动。

附录

附录A 详细API文档

PC端：

序号	接口地址	请求方法	接口描述
1	/testPaper/page	POST	试卷分页查询
2	/testPaper/save/update	POST	添加/修改试卷
3	/testPaper/get/{id}	GET	根据ID获取试卷
4	/testPaper/remove	POST	删除试卷
5	/testQuestions/page	POST	试题分页查询
6	/testQuestions/save/update	POST	添加/修改试题
7	/testQuestions/get/{id}	GET	根据ID获取试题
8	/testQuestions/remove	POST	删除试题
9	/testQuestions/all/testPaper/{id}	GET	获取指定试卷所有试题
10	/student/page	POST	学生分页查询
11	/student/save/update	POST	添加/修改学生信息
12	/student/get/{id}	GET	根据ID获取学生信息
13	/student/remove	POST	删除学生信息
14	/teachers/page	POST	教师分页查询
15	/teachers/save/update	POST	添加/修改教师信息
16	/teachers/get/{id}	GET	根据ID获取教师信息

序号	接口地址	请求方法	接口描述
17	/teachers/remove	POST	删除教师信息
18	/questions/page	POST	心理试题分页查询
19	/questions/save/update	POST	添加/修改心理试题
20	/questions/get/{id}	GET	根据ID获取心理试题
21	/questions/remove	POST	删除心理试题
22	/records/page	POST	测试记录分页查询
23	/records/save/update	POST	添加/修改测试记录
24	/records/get/{id}	GET	根据ID获取测试记录
25	/records/remove	POST	删除测试记录
26	/manage/login/tea	POST	教师登录
27	/manage/login	GET	获取登录信息
28	/manage/login/logout	GET	用户登出
29	/healthy/echarts/{id}	GET	获取健康图表数据
30	/upload/file	POST	文件上传接口

移动端:

序号	接口地址	请求方法	接口描述
1	/notice/findList	POST	公告列表查询
2	/knowledge/getIndex/{count}	GET	首页最新资讯查询
3	/knowledge/findList	POST	知识库多条件查询
4	/types/findList	POST	知识分类查询
5	/testPaper/findList	POST	试卷列表查询
6	/student/login	POST	学生登录
7	/student/save/update	POST	学生信息保存/更新
8	/student/findList	POST	学生信息查询
9	/testQuestions/allQuestion	POST	获取试卷全部题目
10	/records/saveGrade	POST	考试成绩保存
11	/records/findList	POST	考试记录查询
12	/teachers/findList	POST	教师列表查询

序号	接口地址	请求方法	接口描述
13	/healthy/echarts1/{userId}	GET	健康数据图表查询

附录B 数据库表结构

healthy（健康数据表）此表记录学生的健康信息，包括身高、体重、视力和多项健康指标，以及AI生成的健康建议和录入时间，供健康管理分析和历史记录留存。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	数据编号
2	stu_id	bigint	20	否	学生编号
3	height	double	-	否	身高
4	weight	double	-	否	体重
5	eyesight	varchar	255	否	视力
6	ulse_rate	varchar	255	否	脉率
7	blood_pressure	varchar	255	否	血压
8	heart_rate	varchar	255	否	心率
9	breath	varchar	255	否	呼吸
10	blood_sugar	varchar	255	否	血糖
11	blood_lipid	varchar	255	否	血脂
12	month_day	date	-	否	月份
13	create_time	datetime	-	否	录入时间
14	content	text	-	否	AI建议

knowledge（科普知识管理表）此表记录健康相关的科普知识内容，包括所属分类、标题、封面图片及详细内容，用于在平台上展示健康知识。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	科普知识编号
2	banner_img	varchar	255	否	封面图片
3	type_id	bigint	20	否	所属分类
4	content	text	-	否	内容
5	info	varchar	255	否	简介

序号	列名	数据类型	长度	主键	说明
6	create_time	datetime	-	否	创建时间
7	title	varchar	255	否	标题

manage（管理员表）此表存储管理员的账号信息，包括用户名、密码、照片及姓名，用于管理员的系统身份验证与功能操作。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	管理员编号
2	user_name	varchar	50	否	用户名
3	pass_word	varchar	200	否	密码
4	photo_img	varchar	255	否	图片
5	name	varchar	50	否	名称

notice（公告管理表）此表记录校园平台上的公告信息，包括标题、内容、封面图片以及公告状态，用于在系统中展示。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	公告编号
2	title	varchar	255	否	标题
3	notice_img	varchar	255	否	封面图片
4	content	text	-	否	内容
5	info	varchar	255	否	简介
6	status	int	11	否	状态[0：隐藏，1：正常]
7	create_time	datetime	-	否	创建时间

questions（题目表）此表存储心理健康测试中的题目内容及选项，包含每个选项的分数，用于健康测试模块。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	题目编号
2	title	text	-	否	题目标题
3	option_a	varchar	255	否	选项A

序号	列名	数据类型	长度	主键	说明
4	option_b	varchar	255	否	选项B
5	option_c	varchar	255	否	选项C
6	option_d	varchar	255	否	选项D
7	grade_a	varchar	255	否	答案A分数
8	grade_b	varchar	255	否	答案B分数
9	grade_c	varchar	255	否	答案C分数
10	grade_d	varchar	255	否	答案D分数
11	teacher_id	bigint	20	否	老师编号
12	create_time	datetime	-	否	创建时间

records（建档信息表）此表记录学生的测试档案信息，包括学生编号、试卷编号、测试成绩以及对应的评估结果，供健康管理参考。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	档案编号
2	stu_id	bigint	20	否	学生编号
3	test_id	bigint	20	否	试卷编号
4	tea_id	bigint	20	否	教师编号
5	grade	varchar	255	否	考试分数
6	create_time	datetime	-	否	创建时间
7	info	varchar	255	否	对应结果

student（学生表）此表存储学生的基本信息，包括姓名、年龄、性别、联系方式等，用于标识和管理学生用户。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	主键ID
2	user_name	varchar	100	否	用户名（select）
3	pass_word	varchar	255	否	密码
4	head_img	varchar	255	否	头像（img）
5	nick_name	varchar	100	否	姓名

序号	列名	数据类型	长度	主键	说明
6	create_time	datetime	-	否	创建时间

teachers（教师表）此表存储教师的个人信息和系统使用信息，包括姓名、账号、联系方式等，用于教师管理和登录功能。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	教师编号
2	username	varchar	255	否	账号
3	password	varchar	255	否	密码
4	tea_name	varchar	255	否	姓名
5	tea_age	int	20	否	年龄
6	tea_sex	int	20	否	性别
7	tea_info	text	-	否	简介
8	tea_tel	varchar	255	否	联系方式
9	tea_img	varchar	255	否	头像
10	create_time	datetime	-	否	创建时间
11	mail	varchar	255	否	邮箱
12	status	int	11	否	状态[0：待审核，1：可用，2：停用]

test_paper（试卷管理表）此表存储与健康测评相关的试卷信息，包括标题、封面图片和总分等内容，供教师创建和学生使用。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	试卷编号
2	tea_id	bigint	20	否	所属老师
3	title	varchar	255	否	试卷标题
4	img	varchar	255	否	封面图片
5	create_time	datetime	-	否	创建时间
6	total_grade	varchar	255	否	总分
7	info	varchar	255	否	试卷简介

test_questions（试卷题目表）此表将试卷与具体的题目关联，包含试卷编号、题目编号以及教师编号。

序号	列名	数据类型	长度	主键	说明
	id	bigint	20	是	主键ID
2	test_id	bigint	20	否	所属试卷
3	question_id	bigint	20	否	题目编号
4	tea_id	bigint	20	否	老师编号

test_rules（试卷规则管理表）此表存储与试卷测评规则相关的信息，包括分数区间及对应评估结果，用于学生成绩的自动评价。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	规则编号
2	test_id	bigint	20	否	试卷编号
3	min_grade	int	10	否	最低分数
4	max_grade	int	10	否	最高分数
5	info	varchar	255	否	对应结果
6	create_time	datetime	-	否	创建时间
7	tea_id	bigint	20	否	老师编号

types（科普知识分类表）此表记录健康知识的分类信息，包括分类名称和创建时间，用于知识模块数据的归类。

序号	列名	数据类型	长度	主键	说明
1	id	bigint	20	是	分类编号
2	type_name	varchar	255	否	分类名称
3	create_time	datetime	-	否	创建时间

附录C 测试用例清单

用例编号	测试用例名称	测试目标	前置条件	测试步骤	测试结果
TC-001	用户注册功能测试	确认用户可以成功注册并创	系统正常运行，用户未	1. 打开注册页面 2. 输入有效的用	用户成功注册并收到确

用例编号	测试用例名称	测试目标	前置条件	测试步骤	测试结果
		建账户	注册	户名、密码、邮箱 3. 提交注册	认邮件
TC-002	登录功能测试	验证注册用户可通过正确的用户名和密码访问系统	用户已注册且账户已激活	1. 打开登录页面 2. 输入正确的用户名和密码 3. 点击登录按钮	成功登录, 进入用户主页
TC-003	记录健康数据测试	确认用户能够记录并保存健康数据	用户已登录进入系统首页	1. 选择健康数据记录选项 2. 输入相关数据 3. 点击保存按钮	健康数据被成功保存, 显示确认信息
TC-004	数据分析功能测试	验证用户可查看其健康数据的分析结果	用户已登录并有健康数据记录	1. 进入数据分析页面 2. 查看图表及分析结果	正确显示健康数据分析图表
TC-005	心理测试功能测试	确认心理测试功能能够正常运行并输出结果	用户已登录进入系统首页	1. 进入心理测试页面 2. 完成所有心理测试问题 3. 提交测试	系统反馈心理评估结果
TC-006	管理员查看用户信息测试	验证管理员可以查看用户健康和心理信息	管理员已通过身份验证进入管理端	1. 进入用户信息管理页面 2. 查询指定用户信息	正确显示用户的详细健康记录
TC-007	系统安全性测试	测试系统能抵御常见的安全攻击 (如SQL注入)	环境准备充分, 系统处于可攻击的状态	1. 在输入框尝试SQL注入 2. 提交恶意输入	系统未被攻击, 安全性保护有效
TC-008	用户反馈功能测试	确认用户可以成功提交反馈并获得响应	用户已登录进入系统首页	1. 进入反馈页面 2. 输入反馈内容 3. 提交反馈	系统反馈接收成功并提示处理结果