



|  |  |
| --- | --- |
| **项目名称：** | **XX系统** |
| **报告题目：** | **设计报告** |
| **指导老师：** | **尹建伟** |
| **学 院：** | **计算机科学与技术学院** |
| **项目人员：** |  |

**修改历史**

|  |  |  |  |
| --- | --- | --- | --- |
| 日期 | 版本 | 作者 | 修改内容 |
| 2025.5.18 | 1.0 |  | 初稿 |
| 2021.5.20 | 2.0 |  | 终稿 |

目录

[一、 简介 4](#_Toc135658405)

[1.1目的 4](#_Toc135658406)

[1.2 范围 4](#_Toc135658407)

[1.2.1软件名称 4](#_Toc135658408)

[1.2.2软件功能 4](#_Toc135658409)

[1.2.3软件应用 4](#_Toc135658410)

[二、 需求说明 5](#_Toc135658411)

[2.1 功能需求描述 5](#_Toc135658412)

[2.1.1 用户登录 5](#_Toc135658413)

[2.1.2 主页健康数据展示 5](#_Toc135658414)

[2.1.3 训练计划管理 5](#_Toc135658415)

[2.1.4 食谱推荐 5](#_Toc135658416)

[2.2 接口需求描述 6](#_Toc135658422)

[2.3 性能需求描述 7](#_Toc135658423)

[2.3.1 反应时间 7](#_Toc135658424)

[2.3.2 容量与稳定性 7](#_Toc135658425)

[2.3.3 服务器配置最低需求 7](#_Toc135658426)

[2.3.4 数据处理能力 7](#_Toc135658426)

[2.3.5 页面设计 7](#_Toc135658426)

[2.4 安全需求描述 8](#_Toc135658427)

[2.4.1 保密性 8](#_Toc135658428)

[2.4.2 完整性 8](#_Toc135658429)

[2.4.3 约束性 8](#_Toc135658430)

[2.4.4 用户信息安全性 8](#_Toc135658431)

[2.5 环境需求描述 9](#_Toc135658432)

[2.5.1 操作系统 9](#_Toc135658433)

[2.5.2 硬件要求 9](#_Toc135658434)

[2.5.3 浏览器支持 9](#_Toc135658435)

[2.5.4 网络要求 9](#_Toc135658436)

[2.6可维护性需求描述 9](#_Toc135658437)

[三、 系统体系结构图 11](#_Toc135658438)

[四、 数据库设计 12](#_Toc135658439)

[4.1 ER图 12](#_Toc135658440)

[4.2 逻辑结构设计 12](#_Toc135658441)

[五、 关键过程 13](#_Toc135658442)

[5.1 用户登录过程 13](#_Toc135658443)

[5.2 健康数据记录与展示过程 14](#_Toc135658444)

[5.3 训练计划管理过程 14](#_Toc135658445)

[5.4 食谱推荐过程 14](#_Toc135658446)

[5.5 用户信息管理过程 15](#_Toc135658447)

[六、 详细设计 17](#_Toc135658452)

[6.1 接口及依赖关系 17](#_Toc135658453)

[6.1.1外部接口 17](#_Toc135658454)

[6.1.2界面间内部接口 17](#_Toc135658455)

[七、 可靠性及安全性设计 17](#_Toc135658456)

[7.1系统出错设计 17](#_Toc135658457)

[7.1.1 出错信息 17](#_Toc135658458)

[7.1.2补救措施 18](#_Toc135658459)

[7.1.2.1 系统恢复 18](#_Toc135658460)

[7.1.2.2 定时备份 18](#_Toc135658461)

[7.1.2.3 系统维护设计 18](#_Toc135658462)

# 简介

## 1.1目的

## 本报告旨在详细说明FitSeek健康管理系统的设计思路、功能实现、系统架构及相关技术细节，为后续开发、维护和升级提供技术文档支持。

## 1.2 范围

健康管理系统的功能需求点、概要设计以及详细设计。

### 1.2.1软件名称

Fitseek

### 1.2.2软件功能

### FitSeek是一款以体重管理为核心，集健康数据记录、训练计划、食谱推荐和个人信息管理于一体的健康生活应用。

### 1.2.3软件应用

本软件适用于关注健康、希望科学管理体重和生活习惯的广大用户。

# 需求说明

## 2.1 功能需求描述

### 2.1.1 用户登录

用户通过登录页面输入账号信息，进入系统。

### 2.1.2 主页健康数据展示

在主页展示用户的体重、体脂等健康数据，并可进入详情页查看历史记录。

### 2.1.3 训练计划管理

用户可查看、制定和管理个人训练计划，支持训练详情查看。

### 2.1.4 食谱推荐

根据用户健康数据和目标，推荐科学的饮食食谱。

## 2.2 接口需求描述

接口描述如下表：

| ****模块**** | ****接口名称**** | ****请求方式**** | ****URL**** | ****请求参数**** | ****响应参数**** | ****权限要求**** | ****备注**** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ****用户认证**** | 用户注册 | POST | /api/auth/register | - tel（string，必填）：唯一手机号 - name（string，必填）：唯一用户名 - password（string，必填）：≥6 位密码 | - 成功（200）： { "code": 200, "success": true, "user": { "user\_id": "...", ... } } - 失败（400）： { "code": 400, "message": "用户名或手机号已存在" } | 匿名可访问 | 用户名 / 手机号唯一约束，密码需加密存储 |
| ****用户认证**** | 用户登录 | POST | /api/auth/login | - type（int，必填）：0 = 手机号登录，1 = 用户名登录 - value（string，必填）：手机号 / 用户名 - password（string，必填）：密码 | - 成功（200）： { "code": 200, "success": true, "token": "...", "user": { ... } } - 失败（401）： { "code": 401, "message": "未注册/密码错误" } | 匿名可访问 | 登录后返回 JWT Token，有效期 7 天，需存储于请求头 |
| ****训练模块**** | 获取锻炼记录列表 | GET | /api/training/records | - page（可选，int）：页码，默认 1 - limit（可选，int）：每页记录数，默认 20 - type（可选，string）：锻炼类型筛选 | - 成功（200）： { "code": 0, "data": { "total": 25, "records": [...] } } - 失败（401）： { "code": 1001, "message": "未授权" } | 用户需认证 | 支持分页和类型筛选（cardio/strength 等），返回当前用户记录 |
| ****训练模块**** | 添加锻炼记录 | POST | /api/training/records | - workoutType（必填，string）：锻炼类型 - content（必填，string）：内容描述 - duration（必填，int）：时长（分钟） | - 成功（200）： { "code": 0, "data": { "id": 10, "typeName": "有氧", ... } } - 失败（400）： { "code": 1002, "errors": ["参数缺失"] } | 用户需认证 | 自动生成id和时间，typeName根据workoutType映射 |
| ****训练模块**** | 删除锻炼记录 | DELETE | /api/training/records/{id} | - id（必填，int）：记录 ID | - 成功（200）： { "code": 0, "message": "记录删除" } - 失败（404/403）： { "code": 1003/1004, "message": "记录不存在/无权删除" } | 用户需认证 | 仅允许删除当前用户的记录，需校验权限 |
| ****训练模块**** | 获取训练目标 | GET | /api/training/target | 无 | - 成功（200）： { "code": 0, "data": { "weeklyMinutes": 120 } } - 失败（401）： { "code": 1001, "message": "未授权" } | 用户需认证 | 返回当前用户的每周锻炼时长目标 |
| ****训练模块**** | 更新训练目标 | PUT | /api/training/target | - weeklyMinutes（必填，int）：每周时长目标（正整数） | - 成功（200）： { "code": 0, "message": "目标更新", "data": { ... } } - 失败（400）： { "code": 1002, "errors": ["参数非法"] } | 用户需认证 | 需校验weeklyMinutes为正整数，接口幂等性设计 |

## 2.3 性能需求描述

**2.3.1 响应时间**

用户操作响应时间≤0.5秒（单用户在线）。

**2.3.2 容量与稳定性**

支持100个用户并发访问，单表记录量≥1k条时查询性能不下降。

**2.3.3 服务器配置最低要求**

CPU Dual 2GHz, 内存 16G，硬盘5400转

**2.3.4 数据处理能力**

至少支持10000条用户的身高、体重数据，100个运动训练计划数据，30000条体重、体脂率记录，100000条运动、饮食的存储

**2.3.5 页面设计**

页面设计应简洁直观，布局合理，突出重点内容。操作简单，方便用户上手。

## 2.4 安全需求描述

​

**2.4.1 保密性**​

* 用于身份验证的用户名和密码应防止未经授权的用户访问系统​
* 应构建访问控制以防止合法用户非法使用系统资源​
* 在用户登录期间，应该防止SQL注入，密码强制破解和伪造会话入侵​

**2.4.2完整性​**

* 防止非法用户对数据进行无意和恶意的修改、插入、删除、防止数据丢失​
* 防止内部用户对数据进行无意和恶意的修改、插入、删除、防止数据丢失​
* 定期备份数据​

**2.4.3 约束性​**

* 对数据库加上一定的约束，对关键性操作如删除、修改进行限制，防止数据丢失，并对用户进行警示​
* 不同身份所拥有的权限不同，只可以进行自己权限里的操作​

**2.4.4 账户信息安全性**​

* 着重账户信息安全性设计，做到外界人员无法入侵到系统内部​
* 内部人员操作需要留下操作痕迹，使用权管理层可以定期或不定期地稽查系统

## 2.5 环境需求描述

### 2.5.1 操作系统

### 服务器端：Windows10+

### 移动端: Android 13+

### 2.5.2 硬件要求

推荐使用具有高性能和较大显示屏幕的设备，最少内存大小应至少为2GB，并确保拥有足够的存储空间来管理数据，建议不少于 32GB 存储容量。

### 2.5.3 浏览器支持

### Chrome, FireFox或新版Edge浏览器，支持ES6语法。​

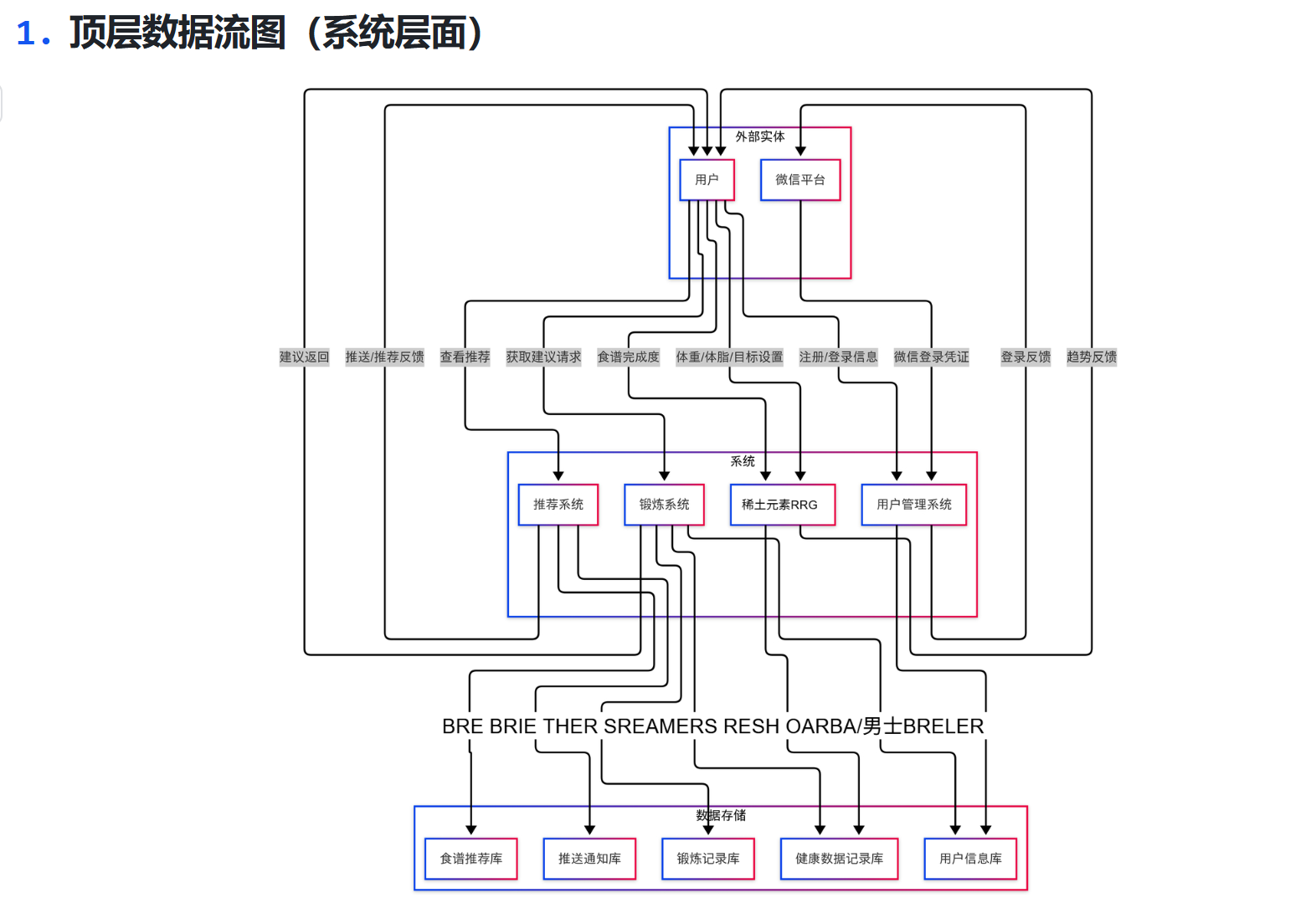
### 2.5.4 网络要求

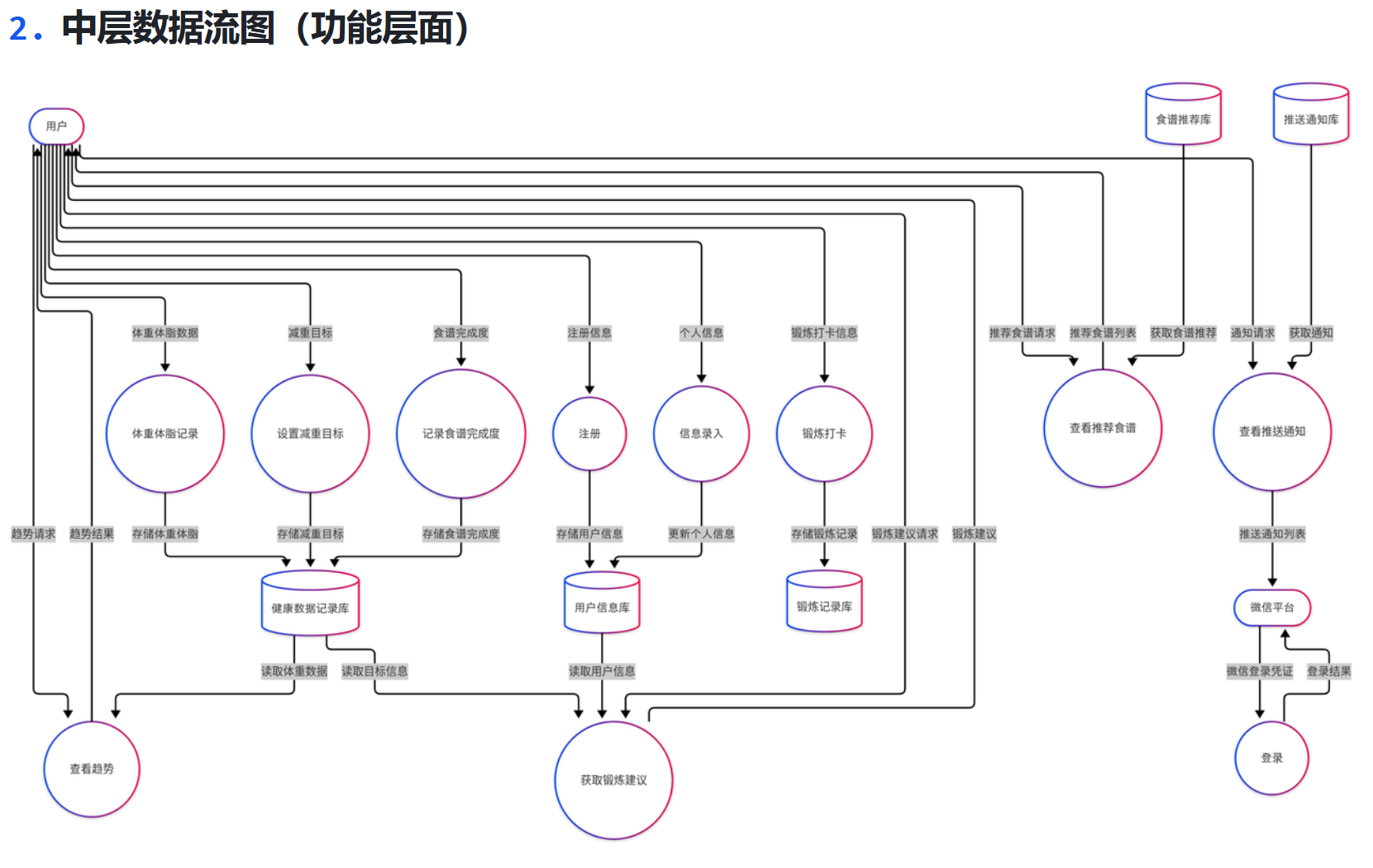
需要稳定的互联网连接才能确保程序正常运行。使用移动网络时，请注意控制流量。另外，在无法建立网络连接的情况下，该系统需要通过缓存数据实现离线功能。

## 2.6 可维护性需求描述

代码接口清晰，采用模块化开发。便于后续维护和功能扩展。

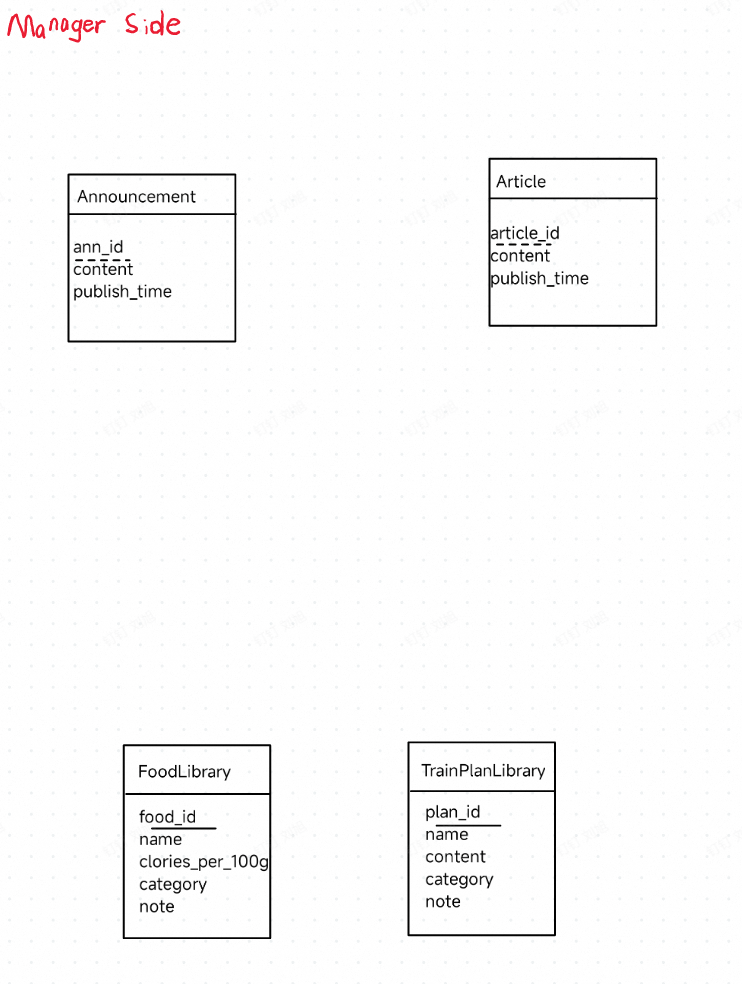
# 三、系统体系结构图



**四、数据库设计**

## 4.1 ER图

## 



## 4.2 逻辑设计结构

### 4.2.1 **实体表设计**

#### **1. 用户信息表（UserProfile）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| user\_id | INT | 主键，自增 | 用户唯一标识 |
| name | VARCHAR(50) | 非空 | 用户名 |
| password\_hash | VARCHAR(128) | 非空 | 加密密码（使用 BCrypt 等算法存储） |
| tel | VARCHAR(20) | 唯一 | 手机号码 |
| type | VARCHAR(10) | 非空（默认 'user'） | 用户类型（user = 普通用户，manager = 管理员） |
| age | INT |  | 年龄 |
| gender | VARCHAR(10) |  | 性别 |
| height | DECIMAL(5,2) |  | 身高（单位：米，如 1.75） |
| weight\_goal | DECIMAL(5,2) |  | 目标体重（单位：公斤） |

#### **2. 健康记录表（FitnessRecord）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| record\_id | INT | 主键，自增 | 健康记录唯一标识 |
| user\_id | INT | 外键（引用 UserProfile.user\_id） | 用户 ID（关联用户信息表） |
| time | DATETIME | 非空 | 记录时间（如 2025-05-18 08:00） |
| value | DECIMAL(5,2) | 非空 | 健康数值（如体重 70.5kg、体脂率 22.3%） |
| note | VARCHAR(200) |  | 备注（如 “晨起空腹称重”） |

#### **3. 饮食记录表（DietRecord）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| record\_id | INT | 主键，自增 | 饮食记录唯一标识 |
| user\_id | INT | 外键（引用 UserProfile.user\_id） | 用户 ID（关联用户信息表） |
| time | DATETIME | 非空 | 饮食时间（如 2025-05-18 12:30） |
| meal\_type | VARCHAR(20) | 非空（早餐 / 午餐 / 晚餐 / 加餐） | 餐型分类 |
| food\_name | VARCHAR(50) | 外键（引用 FoodLibrary.name） | 食物名称（关联食物库表） |
| calories | INT | 非空 | 摄入热量（单位：千卡，如 500 千卡） |
| note | VARCHAR(200) |  | 备注（如 “少油少盐烹饪”） |

#### **4. 训练记录表（TrainRecord）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| record\_id | INT | 主键，自增 | 训练记录唯一标识 |
| user\_id | INT | 外键（引用 UserProfile.user\_id） | 用户 ID（关联用户信息表） |
| time | DATETIME | 非空 | 训练时间（如 2025-05-18 16:00） |
| train\_type | VARCHAR(50) | 外键（引用 TrainPlanLibrary.name） | 训练类型（关联训练计划库表） |
| duration | INT | 非空 | 持续时间（单位：分钟，如 60 分钟） |
| content | VARCHAR(200) |  | 训练内容（如 “慢跑 3 公里”） |
| note | VARCHAR(200) |  | 备注（如 “配速 8 分钟 / 公里”） |

#### **5. 食物库表（FoodLibrary）（管理员维护）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| food\_id | INT | 主键，自增 | 食物唯一标识 |
| name | VARCHAR(50) | 唯一，非空 | 食物名称（如 “鸡胸肉”“燕麦”） |
| calories\_per\_100g | INT | 非空 | 每 100 克热量（单位：千卡，如鸡胸肉 133 千卡 / 100g） |
| category | VARCHAR(20) | 非空（主食 / 蛋白质 / 蔬菜） | 食物类别 |
| note | VARCHAR(200) |  | 备注（如 “优质蛋白质来源”） |

#### **6. 训练计划库表（TrainPlanLibrary）（管理员维护）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| plan\_id | INT | 主键，自增 | 训练计划唯一标识 |
| name | VARCHAR(50) | 唯一，非空 | 计划名称（如 “21 天减脂计划”“增肌力量训练计划”） |
| content | TEXT | 非空 | 计划详情（如每日训练动作及组数） |
| category | VARCHAR(20) | 非空（力量训练 / 有氧运动） | 计划类型 |
| note | VARCHAR(200) |  | 备注（如 “适合新手入门”） |

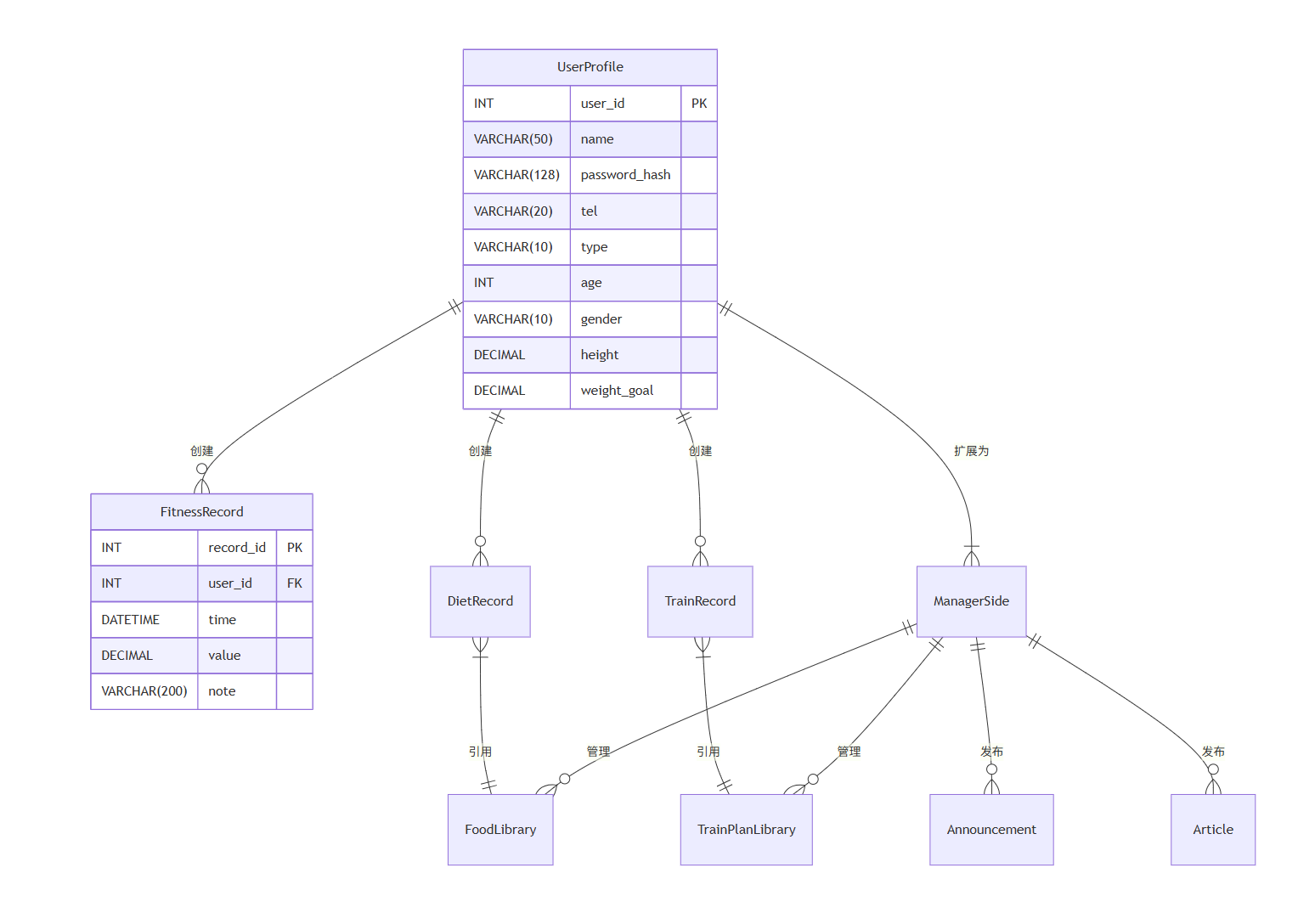
#### **7. 公告表（Announcement）（管理员发布）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| ann\_id | INT | 主键，自增 | 公告唯一标识 |
| content | TEXT | 非空 | 公告内容（如系统升级通知） |
| publish\_time | DATETIME | 非空，默认 CURRENT\_TIMESTAMP | 发布时间（自动生成当前时间） |

#### **8. 文章表（Article）（管理员发布）**

| **字段名** | **数据类型** | **约束条件** | **说明** |
| --- | --- | --- | --- |
| article\_id | INT | 主键，自增 | 文章唯一标识 |
| content | TEXT | 非空 | 文章内容（如《健康饮食指南》） |
| publish\_time | DATETIME | 非空，默认 CURRENT\_TIMESTAMP | 发布时间（自动生成当前时间） |
| category | VARCHAR(20) | 非空（饮食建议 / |  |

## 4.2.2 表关系图



* + ****一对多关系****：如一个用户（UserProfile）对应多条健康记录（FitnessRecord）。
  + ****多对一关系****：如多条饮食记录（DietRecord）可引用同一食物（FoodLibrary）。
  + ****权限继承****：通过 UserProfile 的type字段区分普通用户与管理员，管理员拥有额外维护权限。

### **4.2.3 逻辑设计说明**

****1. 数据分层设计****：

* ****用户业务层****：包含用户信息（UserProfile）及其个人记录（FitnessRecord/DietRecord/TrainRecord）。
* ****基础数据层****：由管理员维护的食物库（FoodLibrary）、训练计划库（TrainPlanLibrary），避免用户重复录入数据。
* ****系统管理层****：公告（Announcement）和文章（Article）用于系统通知和健康知识传播。

1. ****外键约束规则****：

* 所有外键均设置ON DELETE RESTRICT（禁止级联删除），确保基础数据被引用时不可删除。
* 例如：若饮食记录（DietRecord）引用了某食物（FoodLibrary），则该食物在被删除前需先清空相关饮食记录。

1. ****权限控制逻辑****：

* 普通用户（type='user'）只能访问和操作自己的记录（通过user\_id过滤）。
* 管理员（type='manager'）拥有以下权限：
* 增删改查食物库、训练计划库；
* 发布公告和文章；
* 查看所有用户数据（用于系统维护）。

1. ****数据完整性保障****：

* 关键数值字段（如calories、duration）设置非空约束，避免无效数据；
* 日期时间字段（time、publish\_time）统一使用DATETIME类型，确保时间精度。

# 五、关键过程

## 5.1 用户登录过程

## 5.2 健康数据记录与展示过程

## 5.3 训练计划管理过程

## 5.4 食谱推荐过程

## 5.5 用户信息管理过程

# 六、设计详细

## 6.1 接口及依赖关系

**6.1.1 外部接口**

**6.1.2 界面间内部接口**

# 可靠性及安全性设计

**7.1 系统出错设计**

**7.1.1 出错信息**

如登录失败、数据加载异常等，系统友好提示用户。

**7.1.2 补救措施**

**7.1.2.1 系统恢复**

支持断点续传、自动重试。

**7.1.2.2 定时备份**

定期备份用户数据，防止丢失。

**7.1.2.3 系统维护设计**

提供后台管理和日志监控，便于系统维护。