**目录**

1、引言

1.1 编制目的

1.2 词汇表

1.3 参考文献

2、产品概述

3、体系结构设计

3.1 系统概述

3.2 体系结构模型

4、数据库设计

5、界面原型与人机交互设计

6、详细设计

7、WEB性能优化

**1、引言**

1.1 编制目的

本文档详细完成对“行迹”——个人运动社交web应用的体系结构设计和模块的详细设计，达到指导后续软件构造的目的，同时实现和测试人员及用户的沟通。

本文档面向开发人员、测试人员及最终用户而编写，是了解系统的导航。

1.2 词汇表

|  |  |  |
| --- | --- | --- |
| **词汇含义** | **词汇名称** | **备注** |
|  |  |  |
|  |  |  |

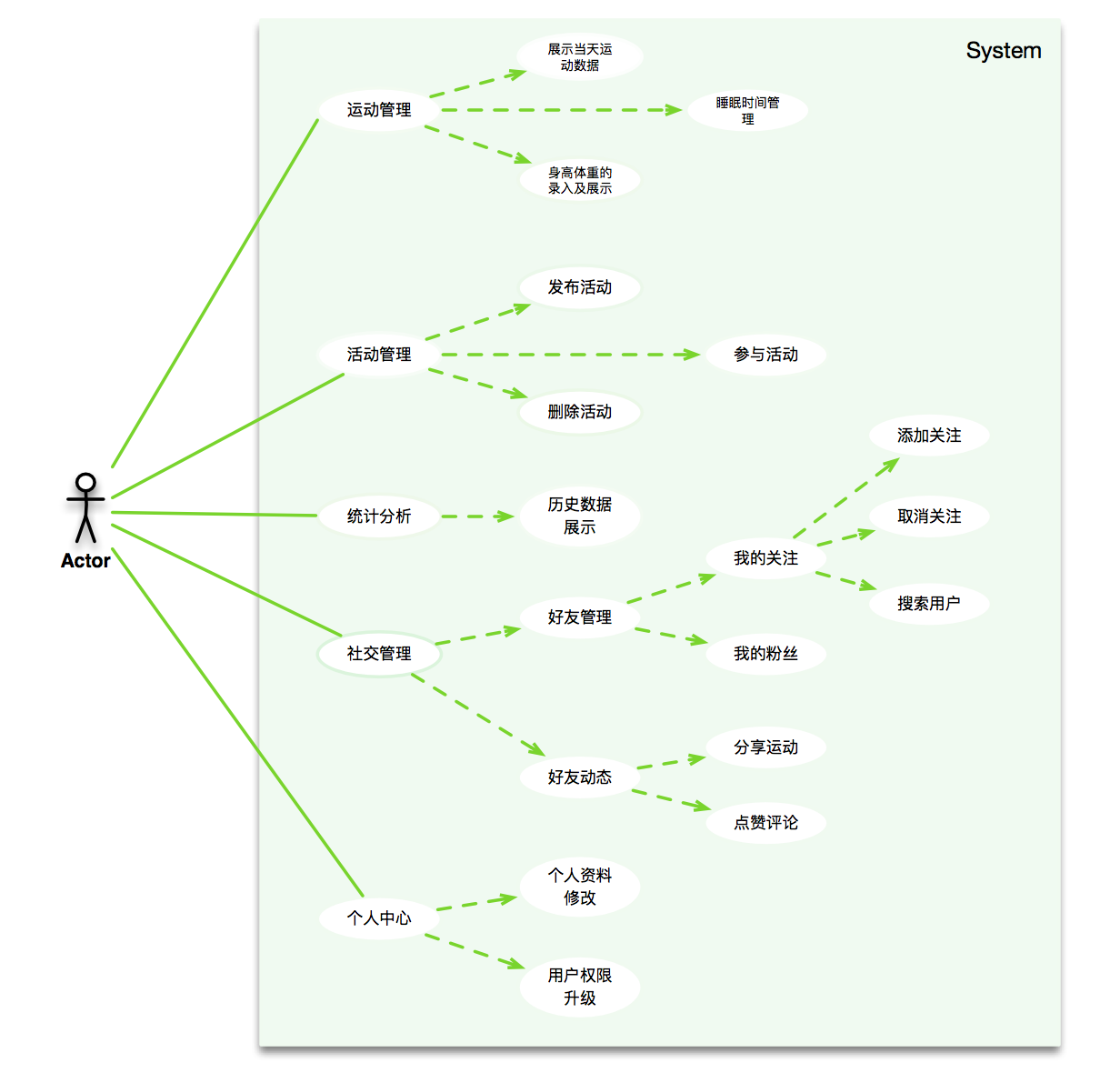
1.3 参考文献

《行迹 需求规格说明书》

**2、产品概述**

“行迹”是一款关注个人身体健康和运动健身，融入了当下流行的互联网社交元素的web应用。“行迹”主要包括以下几个模块：运动健身和个人身体健康的数据采集和统计分析、竞赛和朋友圈等社交活动、（升级VIP用户后）教练量身制定健身计划服务等。

系统用例图如下：

****

**3、体系结构设计**

3.1 系统概述

“行迹”系统构成分为浏览器端（前端）和服务器端（后端）两部分。前端使用Html、CSS、JavaScript语言编写，开发人员自行设计完成，并使用Ajax技术。后端使用PHP语言编写，采用Laravel5.1框架，其中数据库方面使用SQLite 3。

**Laravel开发框架：**

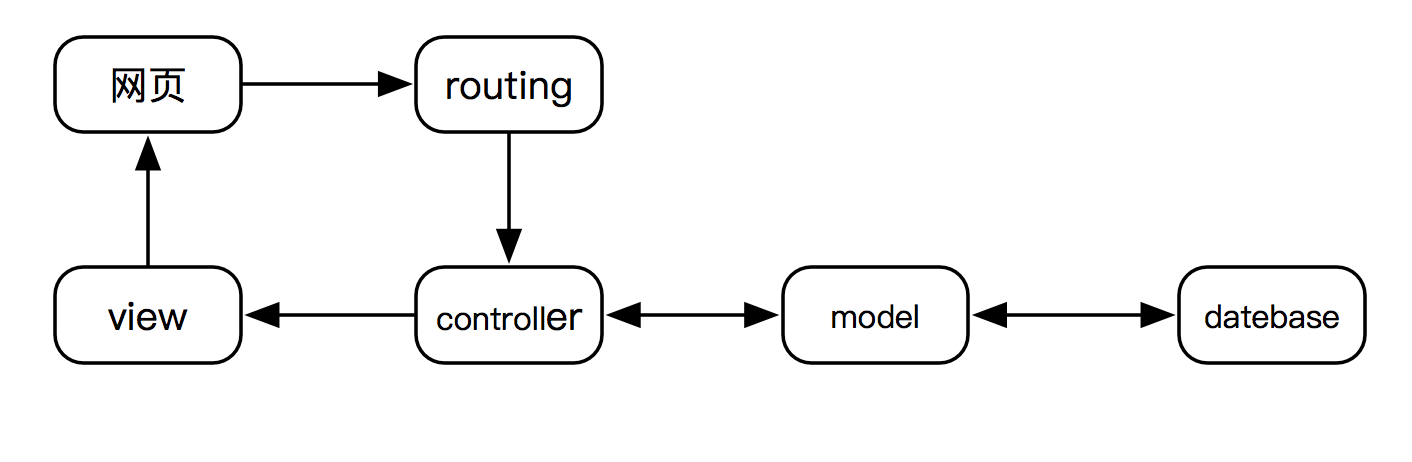
Laravel是一款采用MVC架构的PHP应用开发框架，其应用目录结构如下表所：

|  |  |
| --- | --- |
| 目录 | 描述 |
| app | 包含了应用的核心代码，如Controller、Model等 |
| bootstrap | 包含了少许文件用于框架的启动和自动载入配置，还有一个cache文件夹用于包含框架生成的启动文件以提高性能 |
| config | 包含了应用所有的配置文件 |
| database | 包含了数据迁移及填充文件，以及SQLite数据库存储文件 |
| public | 包含了Laravel框架核心的引导文件index.php，以及任何可以公开的静态资源，如css，Javascript，images等 |
| resources | 包含了视图文件（views）及原生资源文件（LESS、SASS、CoffeeScript），以及本地化文件 |
| storage | 包含了编译过的Blade模板、基于文件的session、文件缓存，以及其它由框架生成的文件 |
| tests | 包含自动化测试 |
| vender | 包含Composer依赖，包含了Laravel源代码和第三方依赖包 |

app文件夹包含了应用的核心代码，其目录结构如下表所示：

|  |  |
| --- | --- |
| 目录/文件 | 描述 |
| /app/Console | 提供进入应用核心的API，包含了所有的Artisan命令 |
| /app/Events | 是放置事件类的地方，事件可以用于通知应用其它部分给定的动作已经发生，并提供灵活的解耦的处理 |
| /app/Exceptions | 包含应用的异常处理器，同时还是处理应用抛出的任何异常的好地方 |
| /app/Http | 提供进入应用核心的API，包含了控制器、过滤器和请求等 |
| /app/Http/Controllers | 包含了用于提供基本的逻辑、数据模型交互以及加载应用程序的视图文件的控制器类 |
| /app/Http/Middleware | 包含了HTTP中间件，对HTTP请求进行过滤 |
| /app/Http/Requests | 包含了对核心类Request的扩展 |
| /app/Http/routes.php | 这是应用的路由文件，其中包含路由规则，告诉Laravel如何将传入的请求连接到路由处理的闭包函数、控制器和操作 |
| /app/Jobs | 是放置队列任务的地方，应用中的任务可以被队列化，也可以在当前请求生命周期内同步执行 |
| /app/Listeners | 包含事件的处理器类，处理器接收一个事件并提供对该事件发生后的响应逻辑 |
| /app/Providers | 包含了服务提供者 |
| /app/\*.php | 在app目录下存储模型（Model），模型是代表应用程序的信息（数据）和操作数据的规则的一些类 |

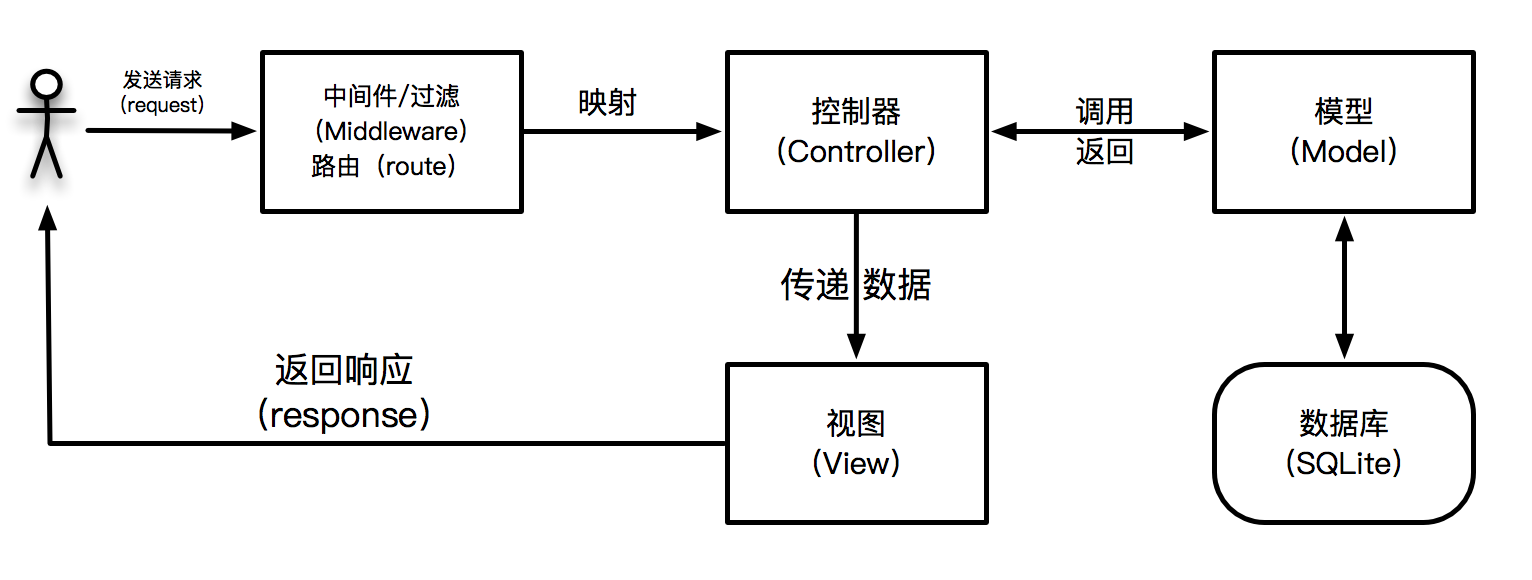
Laravel是model-view-controller（MVC）架构模式，其中Model在app目录下，View在/Resources/views目录下，Controller在/app/Http/Controllers目录下。Laravel的一次响应流程如下图：



当访问一个laravel应用程序时，浏览器发送一个请求，Laravel应用的所有请求入口都是public/index.php文件，所有请求都会被web服务器导向这个文件。接下来Laravel启动内核，Web服务器将请求传递到Laravel的路由引擎。该Laravel路由器接收到请求后，根据配置重定向到相应的控制器类的方法。然后由控制器类接管。在某些情况下，控制器将立即渲染一个视图，这是一个模板，将被转换成HTML并且发送回浏览器。更普遍的动态网站，控制器与模型进行交互，与数据库进行通信。调用模型后,控制器呈现最终视图(HTML、CSS和图像)并返回完整的web页面到用户的浏览器。

3.2 体系结构模型

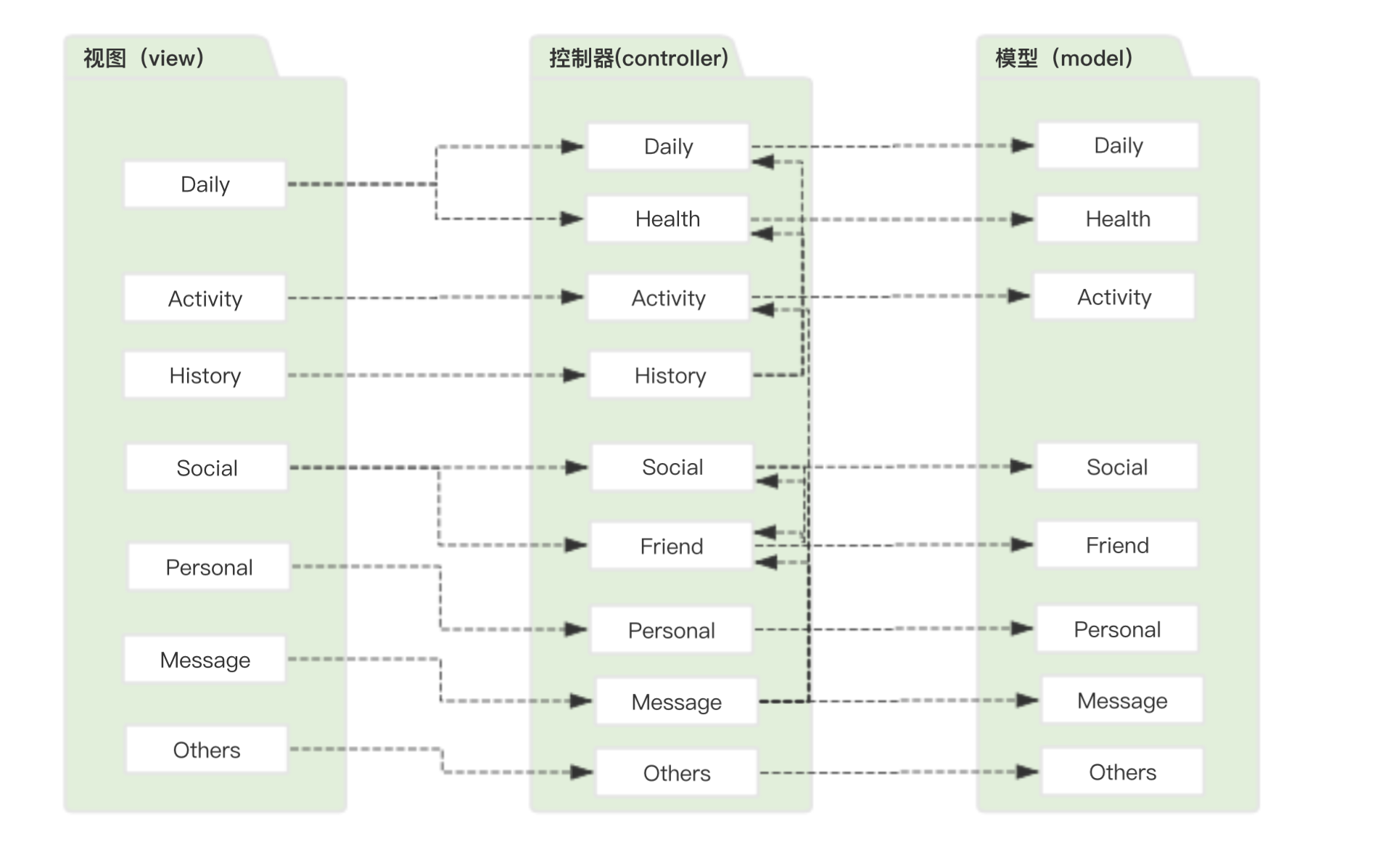
系统的体系结构高层抽象模型如下图所示：



系统暂时只涉及一类用户即所有个人用户（包括普通用户和VIP用户），系统模块表如下：

|  |  |
| --- | --- |
| 模块 | 模块名称 |
| Daily | 运动管理 |
| Health | 身体管理 |
| Activity | 活动（竞赛）管理 |
| History | 统计分析 |
| Social | 朋友圈 |
| Friend | 好友管理 |
| Personal | 个人信息管理 |
| Message | 消息管理 |
| Others | 其他模块（登录、注册、用户权限等） |

系统的开发包图如下：



**4、数据库设计**

系统数据库设计的各表如下所示，其中A\_I为Auto Increment（自增），PK为Primary Key（主键），FK为Foreign Key（外键）

users表（用户表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id | BIGINT, UNSIGNED, A\_I, PK | 用户ID |
| username | VARCHAR, UNIQUE | 用户名 |
| password | VARCHAR | 密码 |
| power | VARCHAR | 普通用户/VIP |
| created\_at | DATETIME | 创建时间 |
| updated\_at | DATETIME | 更新时间 |

user\_info表（个人信息表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id | BIGINT, UNSIGNED, A\_I, PK | 用户ID |
| username | VARCHAR, UNIQUE | 用户名 |
| gender | VARCHAR | 性别 |
| province | VARCHAR | 所在省份 |
| city | VARCHAR | 所在城市 |
| location | VARCHAR | 所在地区 |
| blog | VARCHAR | 个人主页 |
| email | VARCHAR | 个人邮箱 |
| birthday | DATE | 生日 |
| img\_url | VARCHAR | 头像URL |
| info\_secret | VARCHAR | 隐私信息是否可见 |
| health\_secret | VARCHAR | 身体数据是否可见 |

user\_goal表（用户目标表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id | BIGINT, UNSIGNED, A\_I, PK | 用户ID |
| goal | TEXT | 我的目标 |
| created\_at | DATETIME | 创建时间 |
| updated\_at | DATETIME | 更新时间 |

user\_plan表（用户近期计划表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id | BIGINT, UNSIGNED, A\_I, PK | 用户ID |
| plan | TEXT | 近期计划 |
| created\_at | DATETIME | 创建时间 |
| updated\_at | DATETIME | 更新时间 |

user\_hobby表（用户兴趣爱好表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id | BIGINT, UNSIGNED, A\_I, PK | 用户ID |
| hobby | VARCHAR | 兴趣爱好 |

sport\_data表（运动数据表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id | BIGINT, UNSIGNED, A\_I, PK | 用户ID |
| heat | DOUBLE | 燃烧的卡路里数 |
| step | INT | 已走补数 |
| distance | DOUBLE | 运动距离 |
| created\_at | DATETIME | 录入数据时间 |

sleep\_data表（睡眠数据表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id | BIGINT, UNSIGNED, A\_I, PK | 用户ID |
| start\_time | DATETIME | 睡眠开始时间 |
| end\_time | DATETIME | 睡眠结束时间 |
| sleep\_time | TIME | 睡眠时长 |
| sleep\_level | INT | 睡眠质量等级 |
| created\_at | DATETIME | 创建时间 |

health\_data表（身体数据表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id |  |  |
| username |  |  |
| password |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

activity\_data表（活动信息表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id |  |  |
| username |  |  |
| password |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

activity\_user表（活动参与者表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id |  |  |
| username |  |  |
| password |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

social\_share (用户动态表)

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id |  |  |
| username |  |  |
| password |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

share\_good（动态点赞表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id |  |  |
| username |  |  |
| password |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

share\_comment表（动态评论表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id |  |  |
| username |  |  |
| password |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

social\_fans表（粉丝关系表）

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 解释 |
| id |  |  |
| username |  |  |
| password |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |