

“伙伴”系统设计文档

2015软件学院web作业



2015-10-20

洪传旺

目录

[更新历史 2](#_Toc434937917)

[一、引言 3](#_Toc434937918)

[**1.1编制目的** 3](#_Toc434937919)

[**1.2词汇表** 3](#_Toc434937920)

[**1.3参考资料** 3](#_Toc434937921)

[二、架构设计 4](#_Toc434937922)

[2.1 网页原型设计 4](#_Toc434937923)

[2.2 设计思想 8](#_Toc434937924)

[2.3 处理流程 8](#_Toc434937925)

[三、详细设计 12](#_Toc434937926)

[3.1 数据库设计 12](#_Toc434937927)

[3.2 控制器设计 17](#_Toc434937928)

[3.2.1 用户信息管理 17](#_Toc434937929)

[3.2.2 用户管理 19](#_Toc434937930)

[3.2.3 活动管理 21](#_Toc434937931)

[3.2.4 建议管理 23](#_Toc434937932)

[3.2.5 健康管理 26](#_Toc434937933)

[3.3 用户界面 29](#_Toc434937934)

[四、web性能优化手段 35](#_Toc434937935)

# 更新历史

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 变更原因 | 版本号 |
| 洪传旺 | 2015/10/14 | 最初草稿（创建文档模板） | V1.0 |
| 洪传旺 | 2015/11/10 | 正式版（修改不足、添加web性能优化） | V2.0 |

# 一、引言

**1****.1编制目的**

本报告详细完成对“伙伴”系统的概要设计，达到指导详细设计和开发的目的，同时实现和测试人员及用户的沟通

**1.****2词汇表**

|  |  |  |
| --- | --- | --- |
| 词汇名称 | 词汇含义 | 备注 |
| \_view | 表示某视图 | 与thinkphp框架同 |
| \_model | 表示某模块 | 与thinkphp框架同 |
| \_controller | 表示某控制器 | 与thinkphp框架同 |

**1.****3参考资料**

1. IEEE Std 1016-1998 《IEEE Recommended Practice for Software Design Descriptions》

2.《“伙伴”需求文档V1.0》

3.丁二玉，刘钦.计算与软件工程（卷二）[M]机械工业出版社，2012：134—182

# 二、架构设计

## 2.1 网页原型设计

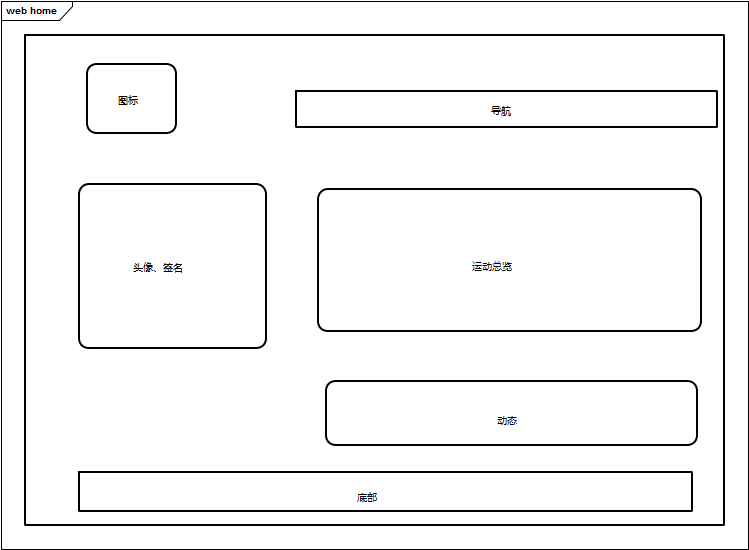


图1 首页

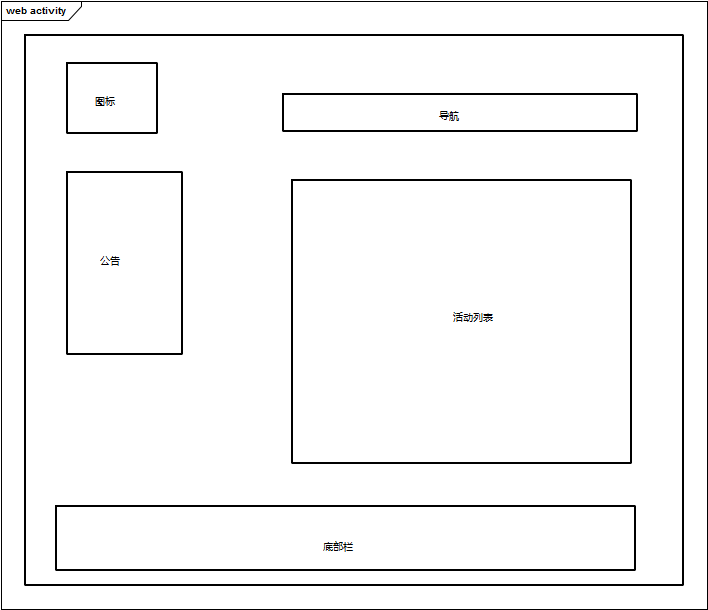


图2 活动管理

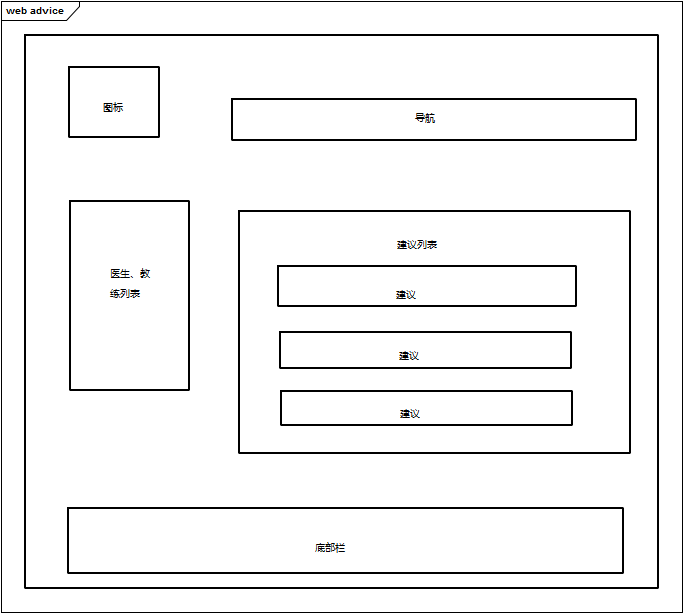


图3 建议管理

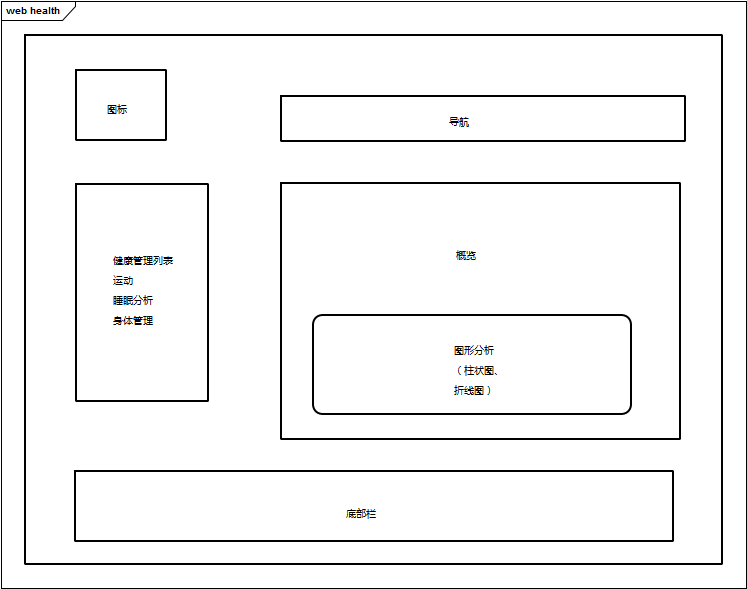


图4 健康管理

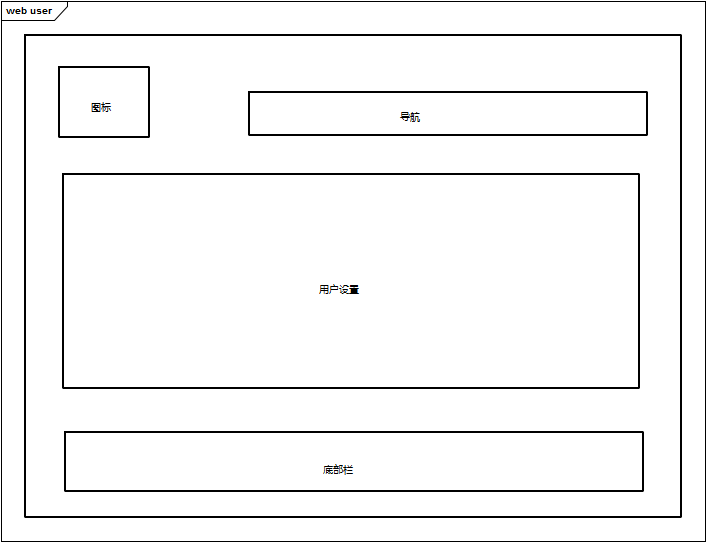


图5 账户设置

## 2.2 设计思想

采用MVC架构进行网站开发，使用thinkphp框架搭建网站，其中前端引入bootstrap框架。

Model将一些细微操作进行封装，供Controller使用。

View主要包含html、css、js等文件，负责页面的显示。

Controller实现与View、Model、数据库的交互。

数据库使用sqlite，具体数据库设计参见详细设计。

## 2.3 处理流程

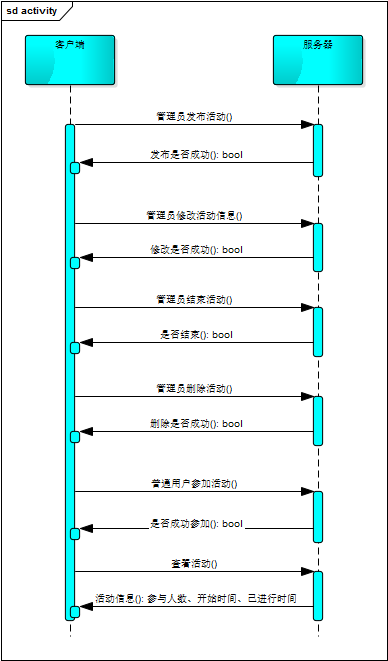


图6 活动管理

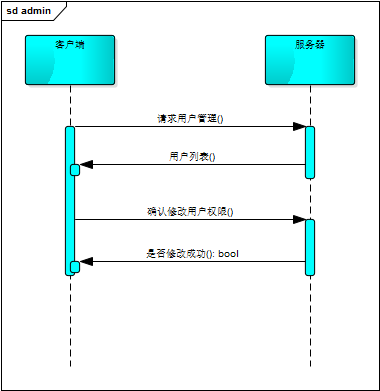


图7 用户管理

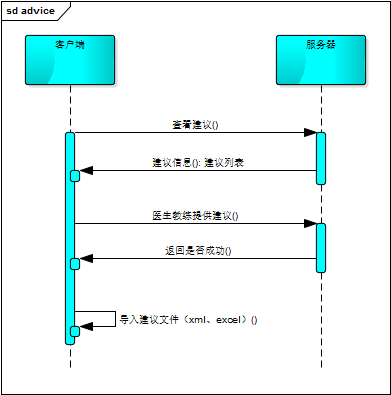


图8 建议管理

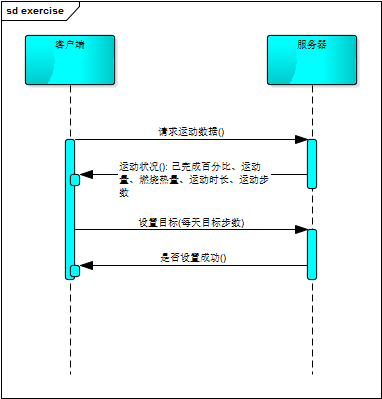


图9 运动分析、状况

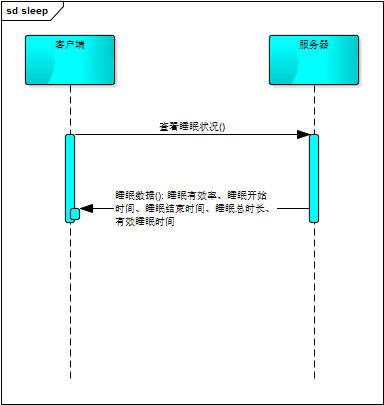


图10 睡眠状况、分析

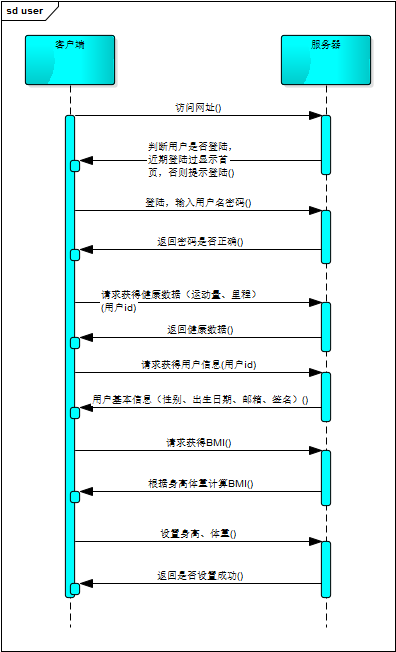


图11 账户管理、登陆

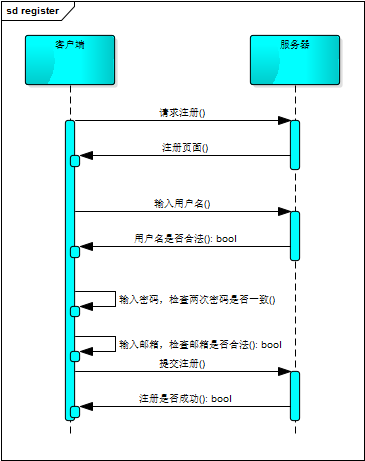


图12 注册

# 三、详细设计

## 3.1 数据库设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表user（用户表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| uid | int | 用户id | 否 | √ |
| id | text | 用户账号 | 否 |  |
| password | text | 用户密码 | 否 |  |
| name | text | 用户昵称 | 否 |  |
| identify | text | 用户身份 | 否 |  |
| sign | text | 个性签名 | 是 |  |
| gender | text | 性别 | 否 |  |
| email | text | 邮箱 | 是 |  |
| birthday | text | 出生日期 | 是 |  |
| icon\_url | text | 头像地址 | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表advice（建议信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| aid | int | 建议id | 否 | √ |
| qid | int | 问题id | 否 |  |
| content | text | 建议内容 | 否 |  |
| to\_user | int | 提问用户id | 否 |  |
| from\_user | int | 回答用户id | 否 |  |
| time | text | 时间戳 | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表question（提问信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| qid | int | 问题id | 否 | √ |
| uid | int | 提问者id | 否 |  |
| to\_user | int | 询问人id | 是 |  |
| title | text | 问题标题 | 否 |  |
| content | text | 问题内容 | 否 |  |
| time | text | 提问时间戳 | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表relation（用户关系表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| rid | int | 关系id | 否 | √ |
| uid | int | 普通用户id | 否 |  |
| service\_id | int | 提供服务者id | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表base\_info（用户基本信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| uid | int | 用户id | 否 | √ |
| enter\_time | text | 用户加入时间 | 否 |  |
| exercise\_amount | int | 运动步数 | 否 |  |
| exercise\_goal | int | 每天运动目标 | 否 |  |
| slumber\_amount | real | 睡眠总量 | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表health\_info（健康信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| uid | int | 用户id | 否 | √ |
| height | real | 身高 | 是 |  |
| weight | real | 体重 | 是 |  |
| BMI | real | BMI指数 | 是 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表exercise\_info（运动信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| uid | int | 用户id | 否 |  |
| date | text | 运动信息时间 | 否 |  |
| step\_number | int | 当天运动步数 | 否 |  |
| xml\_info | text | 当天运动xml文档 | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表slumber\_info（睡眠信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| uid | int | 用户id | 否 |  |
| date | text | 睡眠信息时间 | 否 |  |
| begin\_time | text | 睡眠开始时间 | 否 |  |
| end\_time | text | 睡眠结束时间 | 否 |  |
| slumber\_time | real | 总睡眠时间（分钟） |  |  |
| xml\_info | text | 睡眠信息xml文档 | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表activity（活动信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| activity\_id | int | 活动id | 否 | √ |
| title | text | 活动标题 | 否 |  |
| content | text | 活动内容 | 否 |  |
| img\_url | text | 活动图片地址 | 是 |  |
| enter\_amount | int | 参与人数 | 否 |  |
| post\_time | text | 发布时间 | 否 |  |
| begin\_time | text | 开始时间 | 否 |  |
| end\_time | text | 结束时间 | 否 |  |
| is\_end | int | 是否已经结束 | 否 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表r\_acticity\_user（建议信息表） | | | | |
| 字段 | **字段类型** | **字段说明** | **是否允许为空** | **是否为PK** |
| rau\_id | int | 活动关系id | 否 | √ |
| acticity\_id | int | 活动id | 否 |  |
| uid | int | 用户id | 否 |  |

## 3.2 控制器设计

分为用户信息管理、用户管理、活动管理、建议管理、健康管理五大控制器。下面讲述控制器的模型图。

### 3.2.1 用户信息管理

#### 3.2.1.1 user\_controller概述

user\_controller承担用户信息的管理功能，包括登陆、注册、查看修改基本信息（昵称、签名、性别、出生日期、邮箱、头像）、查看健康总状况（运动目标完成度、睡眠质量、健康BMI）。

#### 3.2.1.2 整体结构

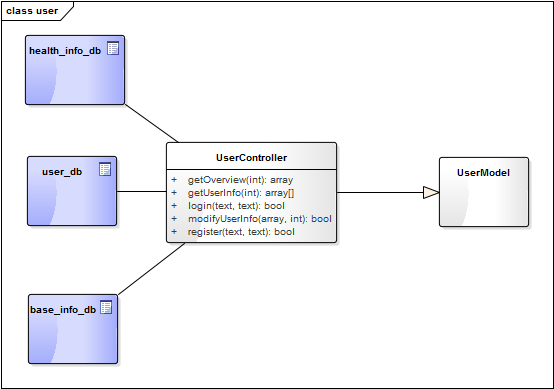


图13 userController整体结构

#### 3.2.1.3 接口规范

|  |  |  |
| --- | --- | --- |
| UserController接口规范 | | |
| login | 语法 | function login($id, $password) |
| 前置条件 | 用户请求登陆 |
| 后置条件 | 返回是否登陆成功 |
| register | 语法 | function register($id, $password) |
| 前置条件 | 用户请求登陆 |
| 后置条件 | 返回是否注册成功 |
| getOverview | 语法 | function getOverview($id) |
| 前置条件 | 用户登录成功，请求获得信息总览 |
| 后置条件 | 返回运动、健康总览 |
| getUserInfo | 语法 | function getUserInfo($id) |
| 前置条件 | 用户请求查看基本信息 |
| 后置条件 | 返回信息数组 |
| modifyUserInfo | 语法 | function modifyUserInfo($id, $userInfo) |
| 前置条件 | 用户登录成功，请求修改基本信息 |
| 后置条件 | 返回修改是否成功 |

#### 3.2.1.5 设计原理

客户端先通过用户名密码登录，获取用户id。之后对于用户信息的管理都通过获取的用户id进行操作，实现客户端与服务器的交互。

用户先从服务器获得用户的基本信息，用户查看后选择可修改的项进行修改，将查看与修改融为一体。

### 3.2.2 用户管理

#### 3.2.2.1 userManage\_controller概述

userManage\_controller承担用户的管理功能，包括修改用户权限、添加删除用户角色、认证医生教练。

#### 3.2.2.2 整体结构

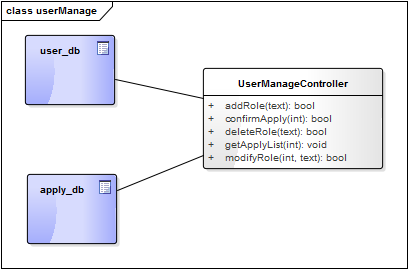


图14 userManageController整体结构

#### 3.2.2.3 接口规范

|  |  |  |
| --- | --- | --- |
| UserManageController接口规范 | | |
| getUserList | 语法 | function getUserList() |
| 前置条件 | 管理员已获得权限 |
| 后置条件 | 返回用户列表 |
| addRole | 语法 | function addRole($identify) |
| 前置条件 | 管理员已获得权限 |
| 后置条件 | 返回是否添加角色成功 |
| deleteRole | 语法 | function deleteRole($identify) |
| 前置条件 | 管理员已获得权限 |
| 后置条件 | 返回是否删除角色成功 |
| modifyRole | 语法 | function modifyRole($id) |
| 前置条件 | 管理员已获得权限 |
| 后置条件 | 返回是否修改用户权限成功 |
| getApplyList | 语法 | function getApplyList () |
| 前置条件 | 管理员已获得权限 |
| 后置条件 | 返回申请修改权限用户列表 |

#### 3.2.2.5 设计原理

管理员先登录网站后台管理，查看所有的用户，选择性的对用户修改、删除、修改权限等。

对于数据的管理只有通过访问后台管理才能操作，保证了网站的安全性。

### 3.2.3 活动管理

#### 3.2.3.1 activity\_controller概述

activity\_controller承担活动管理功能，管理员可以发布、查看、结束、删除活动，普通用户可以参与、查看活动。

#### 3.2.3.2 整体结构

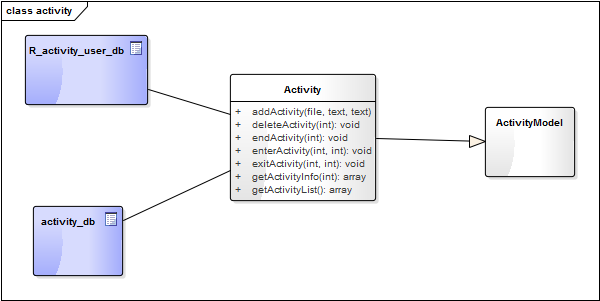


图15 activityController整体结构

#### 3.2.3.3 接口规范

|  |  |  |
| --- | --- | --- |
| ActivityController接口规范 | | |
| addActivity | 语法 | function addActivity ($id, $password) |
| 前置条件 | 管理员获得权限 |
| 后置条件 | 返回是否添加活动成功 |
| deleteActivity | 语法 | function deleteActivity ($id, $password) |
| 前置条件 | 管理员获得权限 |
| 后置条件 | 返回是否删除活动成功 |
| enterActivity | 语法 | function enterActivity ($id) |
| 前置条件 | 管理员获得权限 |
| 后置条件 | 返回是否成功参与活动 |
| exitActivity | 语法 | function exitActivity ($id) |
| 前置条件 | 管理员获得权限 |
| 后置条件 | 返回退出活动成功 |
| getActivityList | 语法 | function getActivityList () |
| 前置条件 | 用户已经经过验证 |
| 后置条件 | 返回修改是否成功 |
| getActivityInfo | 语法 | function getActivityInfo ($aid) |
| 前置条件 | 管理员获得权限 |
| 后置条件 | 返回具体活动信息（包括已参加的人数等） |
| modifyActivity | 语法 | function modifyActivity ($aid, $title, $content) |
| 前置条件 | 管理员获得权限 |
| 后置条件 | 返回修改活动是否成功 |

#### 3.2.3.5 设计原理

用户登录后可查看活动列表并可查看某个活动的具体详情并参加。

管理员登录后台管理后可对活动进行相应的管理，包括添加、修改、删除、结束。

后台管理的机制保证了网站的安全性。

### 3.2.4 建议管理

#### 3.2.4.1 advice\_controller概述

advice\_controller承担建议管理功能，医生、教练用户可以通过普通用户的申请，给普通用户发建议；普通用户可以向医生、教练用户申请绑定，解除绑定，向医生、教练用户提问。

#### 3.2.4.2 整体结构

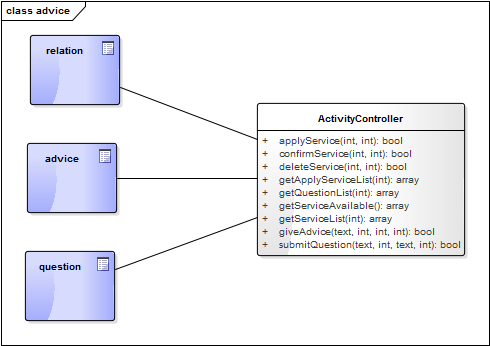


图16 AdviceController整体结构

#### 3.2.4.3 接口规范

|  |  |  |
| --- | --- | --- |
| AdviceController接口规范 | | |
| getServiceAvailable | 语法 | function getServiceAvailable () |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回可申请的医生、教练列表 |
| applyService | 语法 | function applyService ($uid, $service\_id) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回是否绑定医生、教练成功 |
| confirmService | 语法 | function confirmService ($uid, $service\_id) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回是否绑定成功 |
| deleteService | 语法 | function deleteService ($uid, $service\_id) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回是否解除绑定成功 |
| getApplyServiceList | 语法 | function getApplyServiceList ($service\_id) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回申请列表 |
| getServiceList | 语法 | function getServiceList ($uid) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回已绑定医生、教练列表 |
| getQuestionList | 语法 | function getQuestionList ($uid) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回问题列表 |
| submitQuestion | 语法 | function submitQuestion ($uid, $to\_user, $title, $content) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回问题发布是否成功 |
| giveAdvice | 语法 | function giveAdvice ($qid, $from\_user, $to\_user, $content) |
| 前置条件 | 用户已经通过验证 |
| 后置条件 | 返回是否成功 |

#### 3.2.4.5 设计原理

一个用户可以申请多个教练或医生，医生可以通过普通用户的申请成为该用户的绑定医生或教练。

一个普通用户可向管理员提交医生、教练资格申请，管理查看用户申请信息后可选择同意与不同意。

用户发布问题后，教练医生可对该用户提出一些建议。

管理员登录后台管理后可查看普通用户的申请，并选择同意或不同意。

### 3.2.5 健康管理

#### 3.2.5.1 health\_controller概述

health\_controller承担用户的健康管理功能，用户可以查看任意一天的运动状况、睡眠状况、健康状况。

#### 3.2.5.2 整体结构

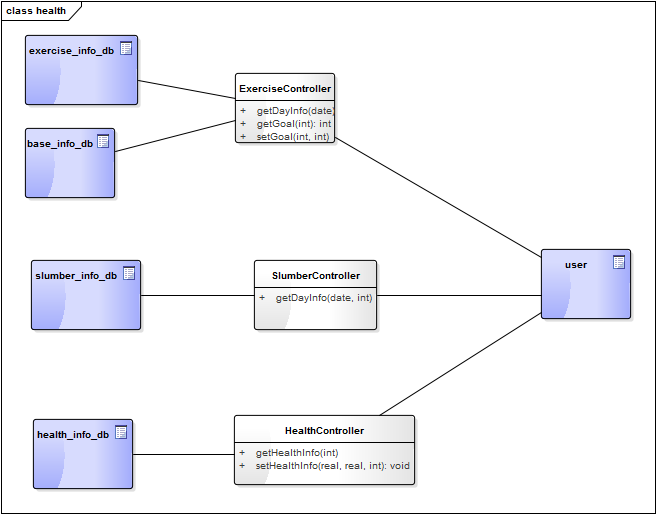


图17 healthController整体结构

#### 3.2.5.3 接口规范

|  |  |  |
| --- | --- | --- |
| ExerciseController接口规范 | | |
| getDayInfo | 语法 | function getDayInfo ($uid, $time) |
| 前置条件 | 用户已经经过验证 |
| 后置条件 | 返回查询日期的运动信息 |
| getGoal | 语法 | function getGoal ($uid) |
| 前置条件 | 用户已经经过验证 |
| 后置条件 | 返回用户设置的运动目标 |
| setGoal | 语法 | function setGoal ($uid) |
| 前置条件 | 用户已经经过验证 |
| 后置条件 | 返回设置目标是否成功 |

|  |  |  |
| --- | --- | --- |
| SlumberController接口规范 | | |
| getDayInfo | 语法 | function getDayInfo ($uid, $time) |
| 前置条件 | 用户已经经过验证 |
| 后置条件 | 返回查询日期的睡眠信息 |

|  |  |  |
| --- | --- | --- |
| HealthController接口规范 | | |
| getDayInfo | 语法 | function getDayInfo ($uid, $time) |
| 前置条件 | 用户已经经过验证 |
| 后置条件 | 返回查询日期的健康信息 |
| setHealthInfo | 语法 | function setHealthInfo ($uid, $height, $weight) |
| 前置条件 | 用户已经经过验证 |
| 后置条件 | 返回是否成功设置身高、体重 |

#### 3.2.5.5 设计原理

原理基本同“用户信息管理”设计原理。

## 3.3 用户界面



图18 首页



图19 身体管理



图20 健身追踪



图21 运动



图22 睡眠分析



图23 活动



图24 我的医生



图25 我的建议

图26 我的提问



图27 用户信息



图28 用户头像

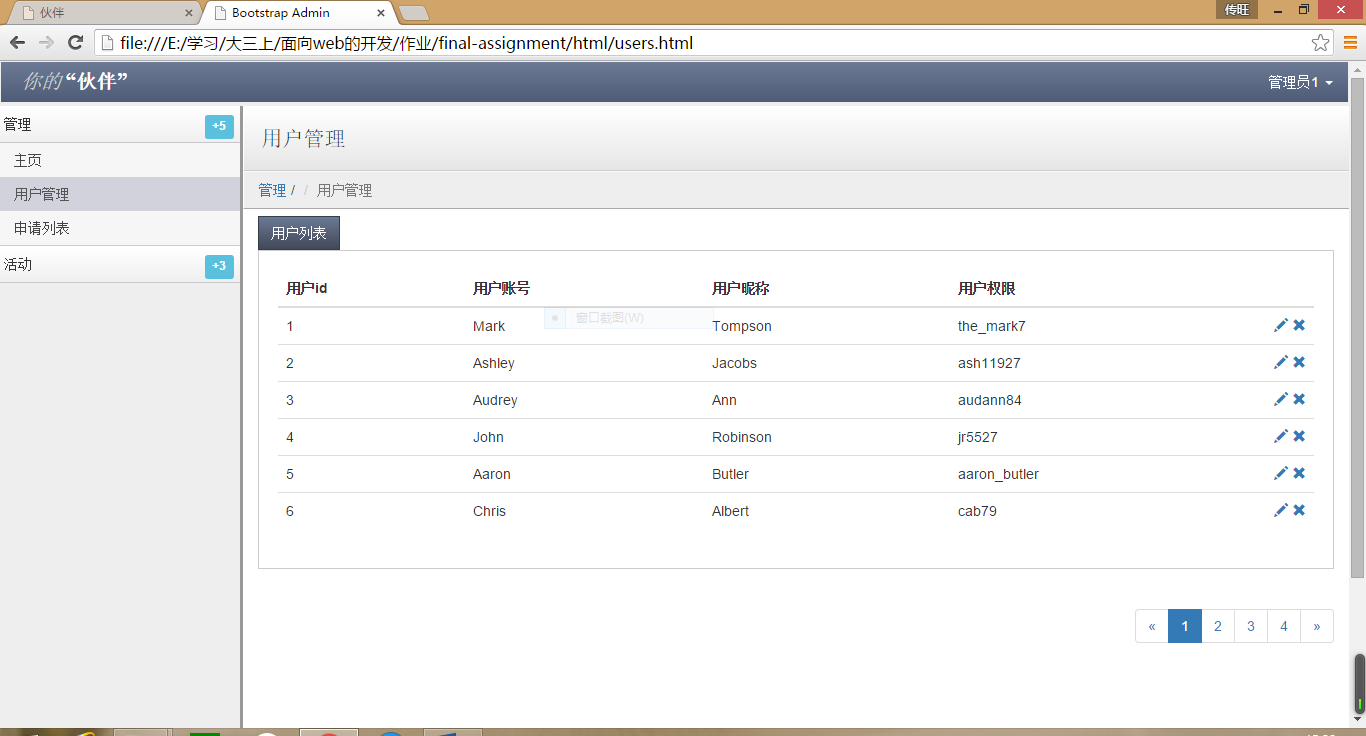


图29 用户管理

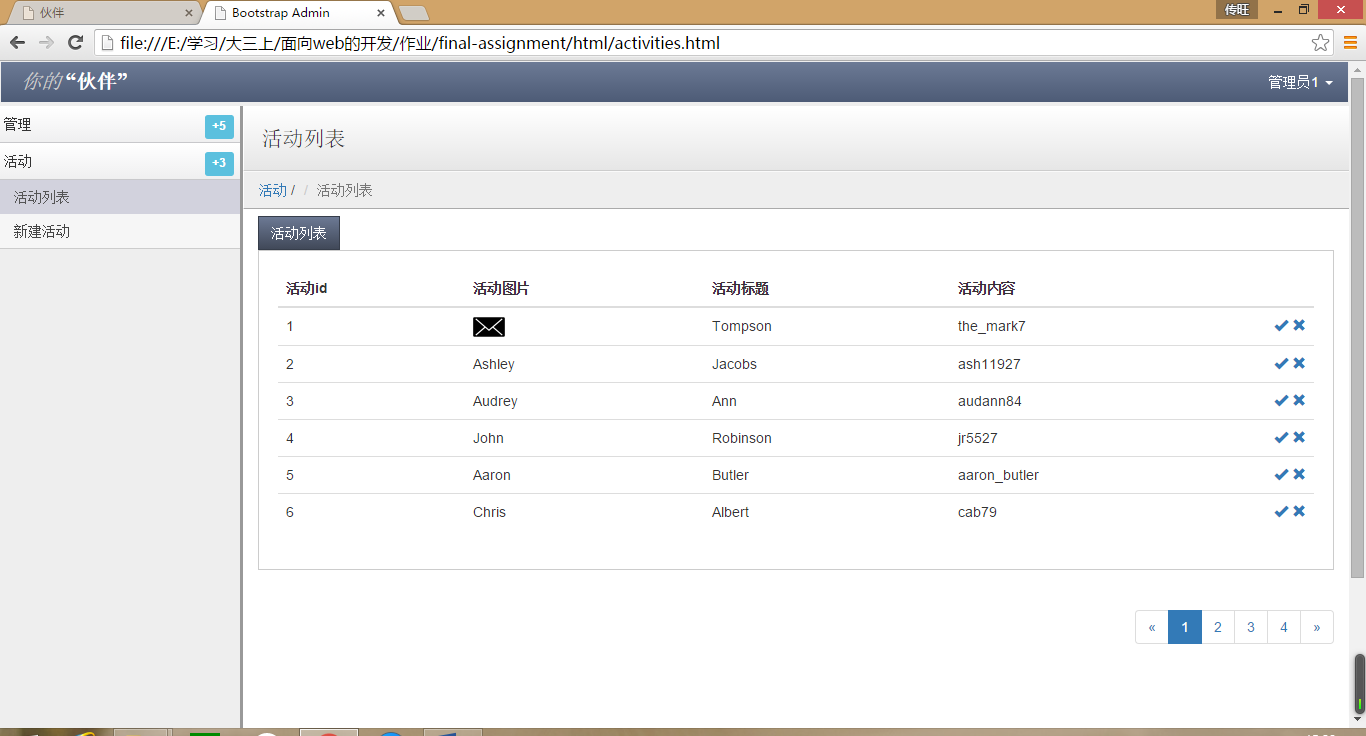


图30 活动管理

# 四、web性能优化手段

1. 尽量减少HTTP请求
   1. 合并脚本跟样式文件，把多个css文件合成一个，把多个js文件合成一个。
   2. CSS Sprites 利用 CSS background 相关元素进行背景图绝对定位,把多个图片合成一个图片。
2. 使用浏览器缓存
   1. 相同的JS、CSS、图片等建议或强制浏览器在本地缓存这些文件。
   2. 在缓存未到期之前，浏览器将直接使用本地缓存文件不会与服务器产生任何通信。
3. 脚本放在底部
4. 样式文件放在页面顶部
5. 使用外部的JS、CSS
6. 精简JS
   1. 精简：从代码中移除不必要的字符以减少其大小。
   2. 混淆：在精简的同时，还会改写代码，函数、变量名被转换成更短的字符串。
7. 精简CSS，从代码中移除不必要的字符以减少其大小。