详细设计文档

杜天蛟141250031

**变更历史**

|  |  |  |  |
| --- | --- | --- | --- |
| 修改人员 | 日期 | 变更原因 | 版本号 |
| 杜天蛟 | 2016-10-28 | 初稿 | 1.0 |

目录

[1引言 2](#_Toc465546132)

[1.1编制目的 2](#_Toc465546133)

[1.2参考资料 2](#_Toc465546134)

[2总体设计 2](#_Toc465546135)

[2.1开发环境 2](#_Toc465546136)

[2.2模块结构 3](#_Toc465546137)

[2.3系统设计模式 3](#_Toc465546138)

[2.4软件架构 3](#_Toc465546139)

[3数据库设计 3](#_Toc465546140)

[4导航设计 5](#_Toc465546141)

[4.1整体导航设计 5](#_Toc465546142)

[4.2活动导航 5](#_Toc465546143)

[4.3好友管理导航 6](#_Toc465546144)

[5详细设计 6](#_Toc465546145)

[5.1登录/注册 6](#_Toc465546146)

[5.2活动管理 8](#_Toc465546147)

[5.3个人管理 9](#_Toc465546148)

[5.4用户管理 10](#_Toc465546149)

[5.5健康管理 11](#_Toc465546150)

[5.6健康管理 12](#_Toc465546151)

[5.6朋友圈 13](#_Toc465546152)

# 1引言

## 1.1编制目的

本文档是Gofit网站的详细设计文档，用于指导后续的软件开发，实现与测试及用户的沟通。

本文档说明的内容可能在项目实施过程中发生变更，但是必须由项目执行者仔细分析最终决定，建立持续有效的版本控制。

## 1.2参考资料

1）IEEE标准

2）Gofit需求文档

# 2总体设计

## 2.1开发环境

本系统采用php作为主要开发语言，服务端主要使用php＋sqlite+Apache，客户端使用html+css+js。用Apache作为服务器，采用sqlite作为后台数据管理系统。

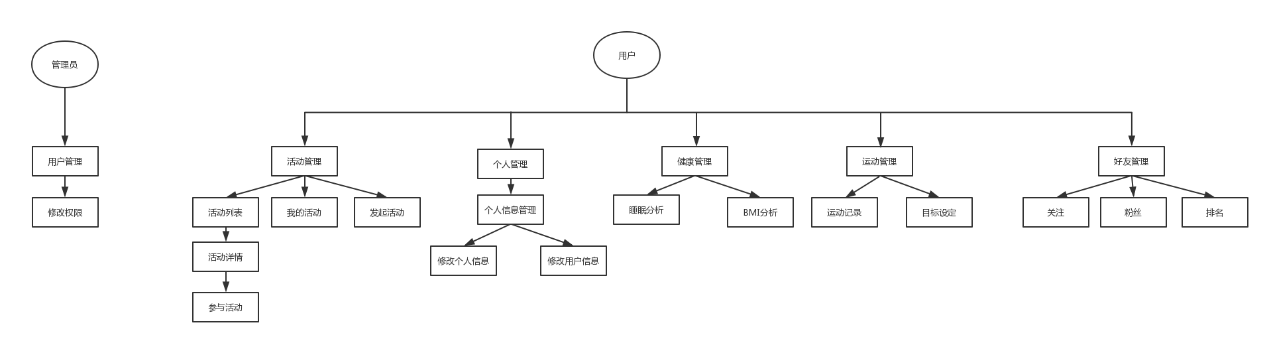
开发环境：winows10

开发工具：phpstorm

测试浏览器：Chrome, Edge

服务器：Apache2

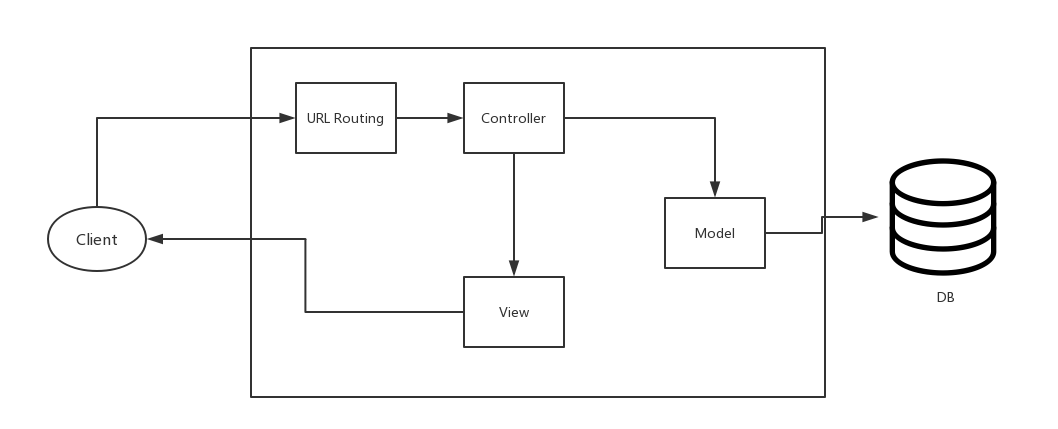
## 2.2模块结构



## 2.3系统设计模式

本系统采用slim框架来创建 RESTful 的 Web 应用，基于MVC设计模式。

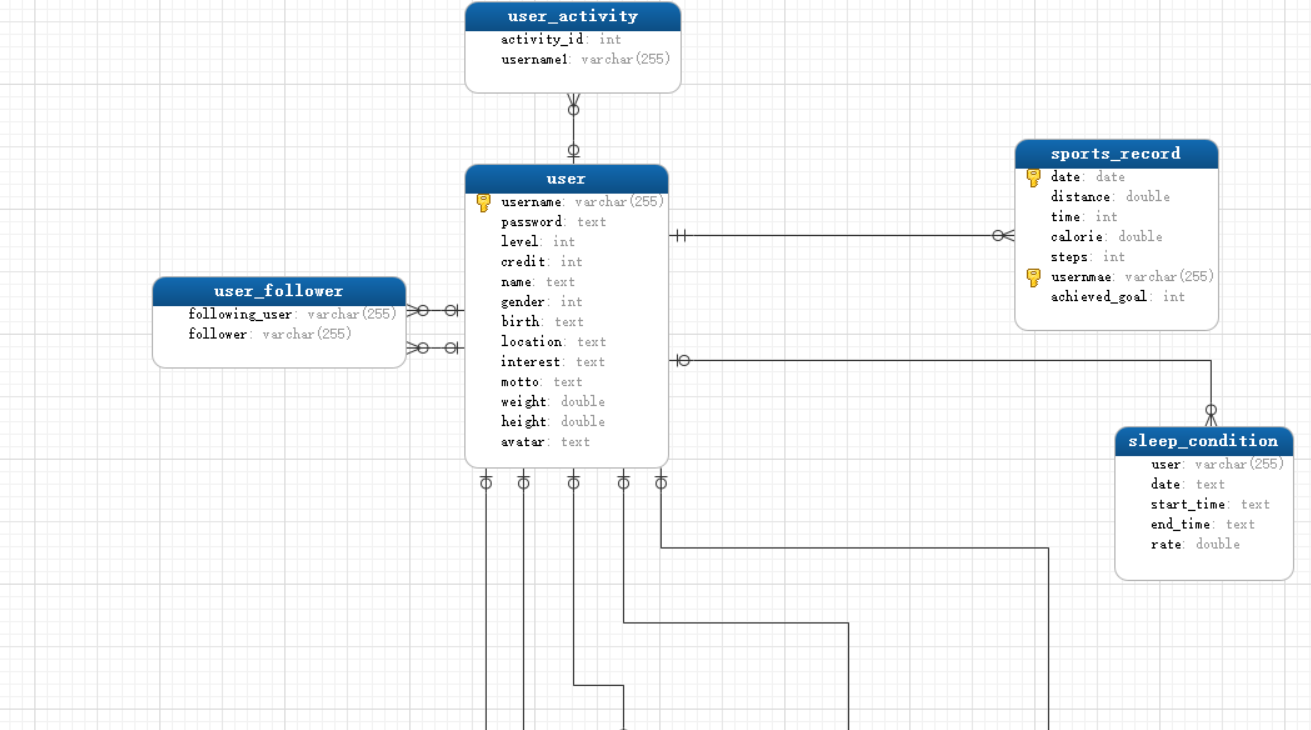
Basic MVC Request Flow

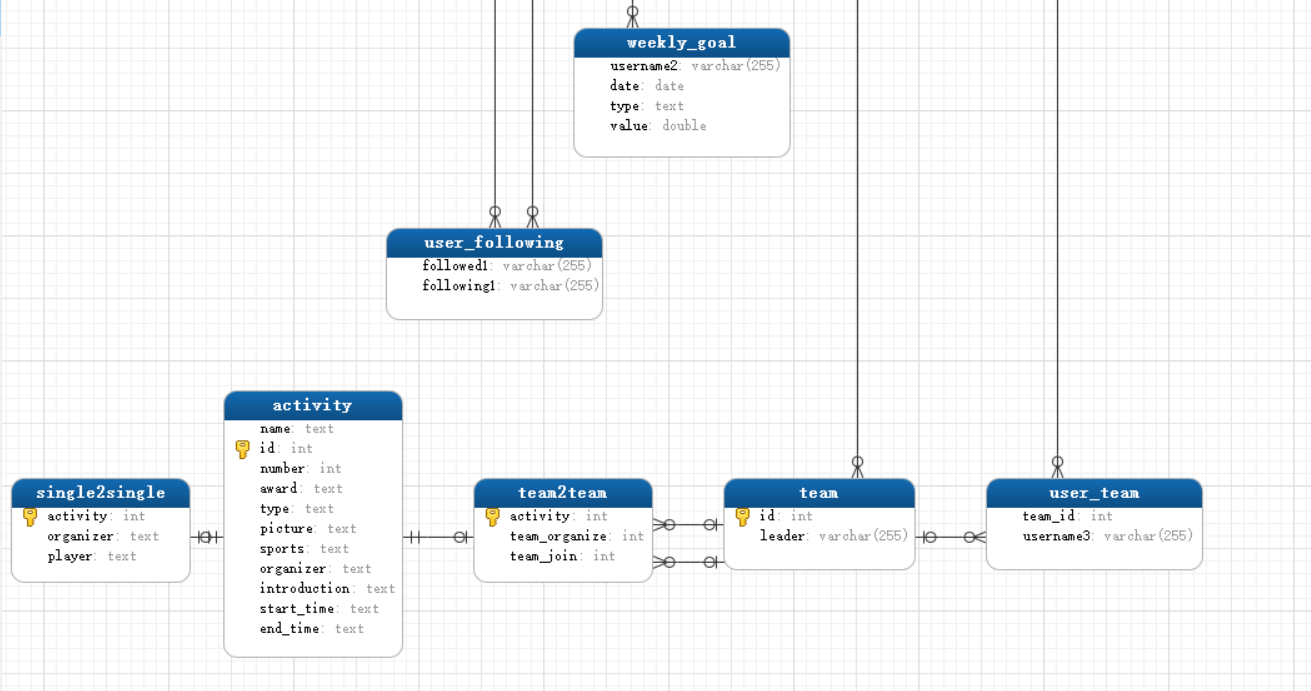


## 2.4软件架构

本系统采用RESTful架构

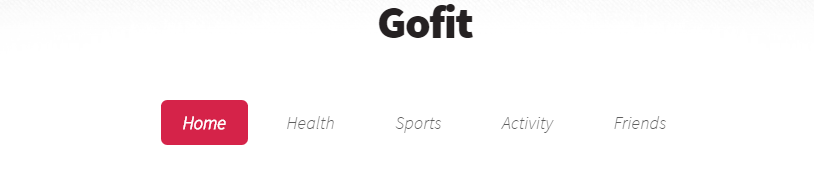
# 3数据库设计



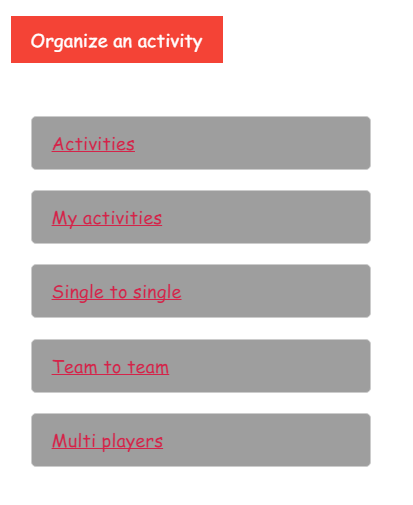


# 4导航设计

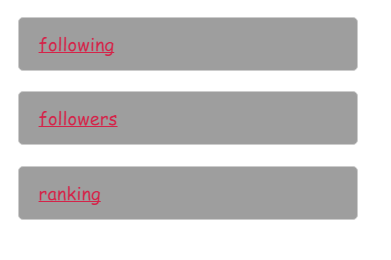
## 4.1整体导航设计



## 4.2活动导航



## 4.3好友管理导航



# 5详细设计

## 5.1登录/注册

#### 5.1.1提供的接口设计

|  |  |  |
| --- | --- | --- |
| 登录界面 | url | /Login |
| 方法类型 | Get |
| 控制器 | AuthController |
| 方法 | getLogin() |
| 参数 | 无 |
| 前置条件 | 用户未登录 |
| 后置条件 | 返回登录界面 |
| 登录请求 | url | /Login |
| 方法类型 | Post |
| 控制器 | AuthController |
| 方法 | postLogin(Request$request) |
| 参数 | 用户名，密码（密文） |
| 前置条件 | 用户未登录 |
| 后置条件 | 返回登录信息，失败返回失败提示信息 |
| 注册界面 | url | /SignUp |
| 方法类型 | Get |
| 控制器 | AuthController |
| 方法 | getRegister() |
| 参数 | 无 |
| 前置条件 | 用户未登录 |
| 后置条件 | 返回注册界面 |
| 注册请求 | url | /SignUp |
| 方法类型 | Post |
| 控制器 | AuthController |
| 方法 | postRegister(Request$request) |
| 参数 | 邮箱，用户名，密码（密文） |
| 前置条件 | 用户未登录 |
| 后置条件 | 返回注册结果 |

#### 5.1.2需要的接口

| 服务名 | 服务 |
| --- | --- |
| User::create | 创建用户并加入数据库 |
| User::find | 查找用户 |
| User::save | 保存用户信息并更新数据库 |

## 5.2活动管理

#### 5.2.1提供的接口设计

|  |  |  |
| --- | --- | --- |
| 活动界面 | url | /Activity |
| 方法类型 | Get |
| 控制器 | ActivityController |
| 方法 | getActivityList() |
| 参数 | 无 |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回活动界面 |
| 活动详情 | url | /Activity |
| 方法类型 | Get |
| 控制器 | ActivityController |
| 方法 | getActivity() |
| 参数 | 活动id |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回活动详情 |
| 参与活动 | url | /Activity |
| 方法类型 | Post |
| 控制器 | GameController |
| 方法 | participateIn(Request$request) |
| 参数 | 活动id |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回活动界面 |
| 查看我的活动 | url | /Activity |
| 方法类型 | Get |
| 控制器 | ActivityController |
| 方法 | getMyActivity() |
| 参数 | 无 |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回我的活动 |
| 组织活动 | url | /organizeActivity |
| 方法类型 | Post |
| 控制器 | GameController |
| 方法 | organizeActivity() |
| 参数 | 用户名Username,活动类型type，活动介绍introduction,活动时间start,end,活动奖励award |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回活动界面 |

#### 5.2.2需要的接口

| 服务名 | 服务 |
| --- | --- |
| Activity::getActivityList | 获得活动列表 |
| Activity::getActivity | 获得活动详情 |
| Activity::participateInActivity | 参与活动 |
| User::getMyActivity | 查看我的活动 |
| User::organizeActivity | 组织活动 |

## 5.3个人管理

#### 5.3.1提供的接口设计

|  |  |  |
| --- | --- | --- |
| 个人信息界面 | url | /SelfInfo |
| 方法类型 | Get |
| 控制器 | UserController |
| 方法 | getUserInfo() |
| 参数 | 无 |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回个人信息界面 |
| 修改个人信息请求 | url | /SelfInfo |
| 方法类型 | Post |
| 控制器 | UserController |
| 方法 | postReset(Request$request) |
| 参数 | 昵称，个性域名，性别，生日，所在地，兴趣，运动宣言 |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回修改结果 |

#### 5.3.2需要的接口

| 服务名 | 服务 |
| --- | --- |
| User::find | 查找用户 |
| User::save | 保存用户信息并更新数据库 |

## 5.4用户管理

#### 5.4.1提供的接口设计

|  |  |  |
| --- | --- | --- |
| 用户权限管理界面 | url | /UserAdmin |
| 方法类型 | Get |
| 控制器 | UserAdminController |
| 方法 | UserAdminIndex() |
| 参数 | 无 |
| 前置条件 | 管理员已登录 |
| 后置条件 | 返回用户权限管理界面 |
| 权限修改请求 | url | /UserAdmin |
| 方法类型 | Post |
| 控制器 | UserAdminController |
| 方法 | modifyUserAdmin(Request$request) |
| 参数 | 用户信息，变更权限 |
| 前置条件 | 管理员已登录 |
| 后置条件 | 保存用户权限 |

#### 5.4.2需要的接口

| 服务名 | 服务 |
| --- | --- |
| User::find | 查找用户 |
| Admin::save | 保存权限修改信息 |

## 5.5健康管理

#### 5.5.1提供的接口设计

|  |  |  |
| --- | --- | --- |
| 个人运动记录 | url | /SportsRecord |
| 方法类型 | Get |
| 控制器 | SportsController |
| 方法 | getSportsRecord() |
| 参数 | username |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回身体管理界面 |
| 运动目标设定请求 | url | /setGoal |
| 方法类型 | Post |
| 控制器 | GoalController |
| 方法 | setSportsGoal() |
| 参数 | 无 |
| 前置条件 | 用户已登录 |
| 后置条件 | 无 |

## 5.6健康管理

#### 5.6.1提供的接口设计

|  |  |  |
| --- | --- | --- |
| 修改身体信息请求 | url | /PersonalCentre |
| 方法类型 | Post |
| 控制器 | BodyController |
| 方法 | modifyBodyInfo(Request$request) |
| 参数 | 身高，体重 |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回身体管理界面 |
| 睡眠分析界面 | url | /PersonalCentre |
| 方法类型 | Get |
| 控制器 | SleepController |
| 方法 | sleepIndex() |
| 参数 | 无 |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回睡眠分析界面 |

#### 5.5.2需要的接口

| 服务名 | 服务 |
| --- | --- |
| User::getBodyRecord | 获得身体信息 |
| User::modifyBodyRecord | 修改身体信息 |
| User::getSleepRecord | 获得睡眠信息 |

## 5.6朋友圈

#### 5.6.1提供的接口设计

|  |  |  |
| --- | --- | --- |
| 关注好友 | url | /addFollowing |
| 方法类型 | Post |
| 控制器 | FriendsController |
| 方法 | addFollowing() |
| 参数 | 好友username |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回查找结果 |
| 查看已关注 | url | /showFollowing |
| 方法类型 | Get |
| 控制器 | FriendsController |
| 方法 | getFollowingList() |
| 参数 | 用户username |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回添加结果 |
| 取消关注 | url | /friends/cancelFollowing |
| 方法类型 | Delete |
| 控制器 | FriendsController |
| 方法 | deleteFriend(Request$request) |
| 参数 | 好友username |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回查找好友界面 |
| 查看朋友圈排名 | url | /getRank |
| 方法类型 | Get |
| 控制器 | FriendsController |
| 方法 | getRank() |
| 参数 | 无 |
| 前置条件 | 用户已登录 |
| 后置条件 | 返回排名信息 |

#### 5.6.2需要的接口

| 服务名 | 服务 |
| --- | --- |
| User::addFollowing | 关注好友 |
| User::showFollowing | 查看已关注 |
| User::deleteFollowing | 取消关注 |
| User::getRankList | 查看朋友圈排名 |