**: 目录**

[READ ME 11](#_Toc504134667)

[路径规则: 11](#_Toc504134668)

[**示例数据都为PHP格式, 不是JS** 12](#_Toc504134669)

[关于前端获取PHP变量: 12](#_Toc504134670)

[前端检测用户是否登陆|注册? 12](#_Toc504134671)

[返回数据格式 12](#_Toc504134672)

[物品卡券Item\_coupon 12](#_Toc504134673)

[获取卡券列表 12](#_Toc504134674)

[带page获取卡券列表 12](#_Toc504134675)

[更新卡券 13](#_Toc504134676)

[生成一张卡券 13](#_Toc504134677)

[发放一张卡券 13](#_Toc504134678)

[发放一大堆卡券 13](#_Toc504134679)

[使用一张卡券 13](#_Toc504134680)

[比赛 14](#_Toc504134681)

[创建比赛 14](#_Toc504134682)

[编辑比赛 14](#_Toc504134683)

[编辑比赛战绩 15](#_Toc504134684)

[保存球队友谊赛信息+比赛战绩数据 15](#_Toc504134685)

[直接创建录入比赛战绩 15](#_Toc504134686)

[比赛列表（有页码） 16](#_Toc504134687)

[比赛列表 16](#_Toc504134688)

[比赛列表（所有数据） 17](#_Toc504134689)

[比赛管理操作 17](#_Toc504134690)

[球队成员报名参加比赛 17](#_Toc504134691)

[提交约赛应战信息 18](#_Toc504134692)

[回复约赛应战 18](#_Toc504134693)

[球队参加比赛申请列表（页码） 18](#_Toc504134694)

[球队参加比赛申请列表 19](#_Toc504134695)

[球队战绩列表（有页码） 19](#_Toc504134696)

[球队战绩列表 19](#_Toc504134697)

[球队战绩列表（所有数据） 20](#_Toc504134698)

[历史对手球队列表（有页码） 20](#_Toc504134699)

[历史对手球队列表 20](#_Toc504134700)

[历史对手球队列表（所有数据） 21](#_Toc504134701)

[比赛战绩-会员关联列表（页码） 21](#_Toc504134702)

[比赛战绩-会员关联列表 21](#_Toc504134703)

[比赛战绩-会员关联列表（所有数据） 22](#_Toc504134704)

[印象标签 22](#_Toc504134705)

[印象标签列表（所有数据 无分页） 22](#_Toc504134706)

[发表印象标签评论（一个标签只能评论一次） 22](#_Toc504134707)

[获取印象评论记录列表 23](#_Toc504134708)

[球队 23](#_Toc504134709)

[创建球队 23](#_Toc504134710)

[编辑球队 23](#_Toc504134711)

[球队列表（有页码） 24](#_Toc504134712)

[球队列表（无页码） 24](#_Toc504134713)

[搜索球队列表（有页码） 24](#_Toc504134714)

[搜索球队列表 25](#_Toc504134715)

[我的球队列表 25](#_Toc504134716)

[球队成员列表 25](#_Toc504134717)

[移除球队成员（只有领队可操作） 25](#_Toc504134718)

[球队成员申请离队（成员自己操作） 26](#_Toc504134719)

[申请加入球队 26](#_Toc504134720)

[回复申请加入球队结果 26](#_Toc504134721)

[创建球队活动 26](#_Toc504134722)

[修改球队活动 27](#_Toc504134723)

[直接创建并录入活动 27](#_Toc504134724)

[球队活动列表（有页码） 27](#_Toc504134725)

[球队活动列表 28](#_Toc504134726)

[球队最新活动记录 28](#_Toc504134727)

[报名参加球队活动 28](#_Toc504134728)

[球队活动-会员关联列表（有页码） 29](#_Toc504134729)

[球队活动-会员关联列表 29](#_Toc504134730)

[球队活动-会员关联列表（无分页） 29](#_Toc504134731)

[球队活动、比赛评论列表（有页码） 30](#_Toc504134732)

[球队活动、比赛评论列表 30](#_Toc504134733)

[发布球队活动、比赛评论 30](#_Toc504134734)

[球队活动、比赛点赞/取消点赞 31](#_Toc504134735)

[获取球队活动、比赛点赞状态 31](#_Toc504134736)

[邀请|申请 Apply 32](#_Toc504134737)

[获取招募|申请列表(page) 32](#_Toc504134738)

[获取招募|申请列表(不带page) 32](#_Toc504134739)

[邀请加入|申请加入,写入一条 32](#_Toc504134740)

[邀请加入|申请加入,修改一条 32](#_Toc504134741)

[关注 Follow 33](#_Toc504134742)

[添加关注/取消关注（暂缺球队） 33](#_Toc504134743)

[我的关注列表（有页码） 33](#_Toc504134744)

[被关注（粉丝）列表（有页码） 34](#_Toc504134745)

[获取与当前关注实体（会员、训练营、班级、球队）的关注状态 34](#_Toc504134746)

[招募Recruitment 34](#_Toc504134747)

[搜索招募 34](#_Toc504134748)

[搜索招募(带page) 34](#_Toc504134749)

[获取招募列表 35](#_Toc504134750)

[新增招募接口 35](#_Toc504134751)

[编辑招募接口 35](#_Toc504134752)

[参与招募 35](#_Toc504134753)

[获取响应人名单(无分页) 35](#_Toc504134754)

[获取响应人名单(page) 36](#_Toc504134755)

[订单Bill 36](#_Toc504134756)

[获取会员订单接口 36](#_Toc504134757)

[编辑|添加订单 36](#_Toc504134758)

[训练营Camp 36](#_Toc504134759)

[搜索训练营 36](#_Toc504134760)

[获取训练营的列表有page 37](#_Toc504134761)

[训练营列表 37](#_Toc504134762)

[更新训练营: 37](#_Toc504134763)

[新建训练营 37](#_Toc504134764)

[获取用户创建的训练营id 37](#_Toc504134765)

[训练营评论 38](#_Toc504134766)

[获取评论列表 38](#_Toc504134767)

[获取热门课程（数据打乱） 38](#_Toc504134768)

[训练营关系camp\_member 38](#_Toc504134769)

[申请成为训练营的某个身份 38](#_Toc504134770)

[训练营人员审核 38](#_Toc504134771)

[训练营人员变更 39](#_Toc504134772)

[查询身份 39](#_Toc504134773)

[获取有教练身份的训练营员工 39](#_Toc504134774)

[自定义获取训练营身份列表 39](#_Toc504134775)

[自定义获取训练营身份列表分页（有页码） 39](#_Toc504134776)

[自定义获取训练营身份列表分页（无页码无分页） 40](#_Toc504134777)

[关注训练营成为粉丝 40](#_Toc504134778)

[解除训练营-会员关系（辞退训练营人员） 40](#_Toc504134779)

[教练Coach 40](#_Toc504134780)

[搜索教练列表 40](#_Toc504134781)

[申请成为教练 41](#_Toc504134782)

[获取教练列表 41](#_Toc504134783)

[获取教练身份相关的训练营 41](#_Toc504134784)

[完善教练信息 41](#_Toc504134785)

[场地Court 41](#_Toc504134786)

[获取场地列表 41](#_Toc504134787)

[分页获取场地（带分页、有页码） 42](#_Toc504134788)

[搜索场地 42](#_Toc504134789)

[更新场地 42](#_Toc504134790)

[创建场地 42](#_Toc504134791)

[获取训练营下的场地库 42](#_Toc504134792)

[把场地添加到自己的库 42](#_Toc504134793)

[不分页获取场地列表 43](#_Toc504134794)

[获取courtcamp多条记录(带page) 43](#_Toc504134795)

[获取courtcamp记录(无page无分页) 43](#_Toc504134796)

[会员Member 43](#_Toc504134797)

[完善会员资料 43](#_Toc504134798)

[提现申请 43](#_Toc504134799)

[编辑个人资料 44](#_Toc504134800)

[注销登陆 44](#_Toc504134801)

[登陆|注册Login 44](#_Toc504134802)

[会员注册 44](#_Toc504134803)

[字段是否被占用 44](#_Toc504134804)

[会员登陆 44](#_Toc504134805)

[学生student 45](#_Toc504134806)

[获取会员学生身份列表 45](#_Toc504134807)

[新增学生身份 45](#_Toc504134808)

[更新学生 45](#_Toc504134809)

[证件Cert 46](#_Toc504134810)

[获取证件列表 46](#_Toc504134811)

[添加证件 46](#_Toc504134812)

[修改证件 46](#_Toc504134813)

[教案Plan 46](#_Toc504134814)

[获取教案列表(翻页) 46](#_Toc504134815)

[获取教案列表(不分页) 46](#_Toc504134816)

[获取教案列表(带页码) 47](#_Toc504134817)

[新建教案 47](#_Toc504134818)

[修改教案 47](#_Toc504134819)

[训练项目 Exercise 47](#_Toc504134820)

[获取列表(默认无分页) 47](#_Toc504134821)

[获取列表(带分页) 47](#_Toc504134822)

[编辑项目 47](#_Toc504134823)

[新增项目 48](#_Toc504134824)

[课程Lesson 48](#_Toc504134825)

[新增课程 48](#_Toc504134826)

[编辑课程 48](#_Toc504134827)

[获取课程 48](#_Toc504134828)

[获取购买了课程但是没有分配班级的学生 49](#_Toc504134829)

[获取毕业学生 49](#_Toc504134830)

[获取购买了课程也分配了班级的学生 49](#_Toc504134831)

[获取购买了课程的没毕业的学生 49](#_Toc504134832)

[审核课程: 49](#_Toc504134833)

[获取热门课程 49](#_Toc504134834)

[课程-学生 lesson\_member 50](#_Toc504134835)

[搜索课程学生带page 50](#_Toc504134836)

[搜索课程学生不带page 50](#_Toc504134837)

[获取课程学生数据带page 50](#_Toc504134838)

[获取训练营下的学生带page(唯一)学生|体验生 50](#_Toc504134839)

[获取与课程|训练营相关的学生|体验生-不带page 50](#_Toc504134840)

[获取训练营相同价格的课程带分页page带搜索 51](#_Toc504134841)

[获取未分配班级的学生列表-带page 51](#_Toc504134842)

[活动Event 51](#_Toc504134843)

[新增活动 51](#_Toc504134844)

[编辑活动 51](#_Toc504134845)

[搜索活动列表 52](#_Toc504134846)

[获取活动列表 52](#_Toc504134847)

[获取班级活动接口 52](#_Toc504134848)

[获取活动列表Page 52](#_Toc504134849)

[审核活动: 52](#_Toc504134850)

[录入活动 53](#_Toc504134851)

[参加|购买活动() 53](#_Toc504134852)

[活动人员event\_member 54](#_Toc504134853)

[获取人员列表(唯一) 54](#_Toc504134854)

[获取与活动相关的会员(page) 54](#_Toc504134855)

[获取与活动相关的会员(无page) 54](#_Toc504134856)

[班级Grade 54](#_Toc504134857)

[注意事项: 54](#_Toc504134858)

[创建班级 55](#_Toc504134859)

[获取班级列表 55](#_Toc504134860)

[获取班级列表（不分页） 55](#_Toc504134861)

[更新班级 56](#_Toc504134862)

[班级单个学生变动(编辑班级可用) 56](#_Toc504134863)

[班级批量学生变动(编辑班级可用) 56](#_Toc504134864)

[获取教练下的班级 57](#_Toc504134865)

[班级-学生关系 grade\_member 57](#_Toc504134866)

[获取数据列表带page 57](#_Toc504134867)

[获取与课程|班级(已分班)|训练营相关的学生|体验生-不带page 57](#_Toc504134868)

[课时Schedule 58](#_Toc504134869)

[判断课时冲突(2个小时内只允许一条数据) 58](#_Toc504134870)

[判断是否有录课权限 58](#_Toc504134871)

[课时审核 58](#_Toc504134872)

[录课 58](#_Toc504134873)

[课时评分 59](#_Toc504134874)

[获取列表有page 59](#_Toc504134875)

[获取开始时间和结束时间的列表带page 59](#_Toc504134876)

[课时会员关系 schedule\_member 59](#_Toc504134877)

[获取训练营下的课程(课时名称唯一) 59](#_Toc504134878)

[获取课时列表(page) 60](#_Toc504134879)

[获取课时,无分页 60](#_Toc504134880)

[消息Message 60](#_Toc504134881)

[获取全部个人消息和系统消息列表 60](#_Toc504134882)

[获取系统消息详情 60](#_Toc504134883)

[获取个人消息详情 60](#_Toc504134884)

[设置消息状态 61](#_Toc504134885)

[删除个人消息 61](#_Toc504134886)

[获取个人消息列表 61](#_Toc504134887)

[一些辅助功能 61](#_Toc504134888)

[获取用户地址Location 61](#_Toc504134889)

[获取用户IPLocation 61](#_Toc504134890)

[获取?? 61](#_Toc504134891)

# READ ME

## 路径规则:

#### url(‘地址表达式’,[‘参数’],[‘URL后缀’],[‘域名’])

如

{:url('模块名/方法名',['参数名'=>'参数值'])}

{:url('index/blog/read','id=5&name=thinkphp')}

以上方法会生成下面的URL地址：

/index.php/blog/5/name/thinkphp.html

我们API:

{:url('api/Member/logout',['参数名'=>'参数值'])}

{:url('api/Member/logout')}

lesson->lessonList->lesson

{:url(,[lesson=>'大热'])}

## **示例数据都为PHP格式, 不是JS**

## 关于前端获取PHP变量:

如果要获取一个详情页的班级表的id字段,那么PHP变量为:{$grandeInfo['id']}或者{$grandeInfo.id}

如果要获取一个列表的字段值,则为:{$vol.字段名}或者{$vol['字段名']}

注意:vol只有在当前列表dom有效,有时候vol也为v

## 前端检测用户是否登陆|注册?

js直接读取"{$memberInfo['id']}", 如果有值,则已经登录/注册

## 返回数据格式

type:json

{['code'=>200,'msg'=>{data}]}

# 物品卡券Item\_coupon

## 获取卡券列表

api: getItemCouponListApi

post:

## 带page获取卡券列表

api: getItemCouponListByPageApi

post:

## 更新卡券

api: updateItemCouponApi

post:

get: item\_coupon\_id

PS:涉及到时间的变量要+s,比如publish\_end 要写成publish\_ends

## 生成一张卡券

api: createItemCouponApi

post:

PS:涉及到时间的变量要+s,比如publish\_end 要写成publish\_ends

## 发放一张卡券

api: createItemCouponMemberApi

post|get: item\_coupon\_id

## 发放一大堆卡券

api: createItemCouponMemberListApi

post|get: item\_coupon\_ids

## 使用一张卡券

api: useItemCoupon

post|get: item\_coupon\_member\_id, item\_coupon\_id

# 比赛

## 创建比赛

api: api/match/creatematch

post: match数据表字段

type（比赛类型） 1友谊赛 2联赛

友谊赛提交内容：

team\_id （友谊赛 主队球队id）

opponent\_id （友谊赛 对手球队id）

match\_time（比赛时间）

home\_team\_color、home\_team\_colorstyle （主队球队球衣颜色）

away\_team\_color、away\_team\_colorstyle（对手球队球衣颜色）

response:

成功 code=>200, msg=>'', data=>（新增数据id）

## 编辑比赛

api: api/match/updatematch

post: match数据表字段

response:

成功 code=>200, msg=>''

## 编辑比赛战绩

api: api/match/updatematchrecord

post: match\_record数据表字段

team\_id （友谊赛 主队球队id）

opponent\_id （友谊赛 对手球队id）

match\_time（比赛时间） 当前时间大于填写的比赛时间 自动更新为“已完成”

home\_team\_color、home\_team\_colorstyle （主队球队球衣颜色）

away\_team\_color、away\_team\_colorstyle（对手球队球衣颜色）

response:

成功 code=>200, msg=>''

## 保存球队友谊赛信息+比赛战绩数据

api: api/match/storefriendlymatchrecord()

post: match表字段（记录比赛主要信息），record[]:match\_record表字段（记录比赛战绩、比赛参赛人员）

response:

成功 code=>200, msg=>'', data=>（新增match表id）

## 直接创建录入比赛战绩

api: api/match/creatematchrecord（同时创建match数据、match\_record数据）

post: match\_record数据表字段

team\_id （友谊赛 主队球队id）

opponent\_id （友谊赛 对手球队id）

match\_time（比赛时间） 当前时间大于填写的比赛时间 自动更新为“已完成”

home\_team\_color、home\_team\_colorstyle （主队球队球衣颜色）

away\_team\_color、away\_team\_colorstyle（对手球队球衣颜色）

response:

成功 code=>200, msg=>'', data=>（新增match表数据id）

## 比赛列表（有页码）

api: api/match/matchlistpage

翻页传参: {:url('api/match/matchlsitpage')}?page=页数

post: match数据表字段

year（选择年份 有传入正确年份才查询创建时间在区间内的数据）；

keyword（搜索关键字，发布比赛的球队名）；

choice\_time（比赛时间选择 1：近7天|2：近15天|3：近1个月）

response:

成功 code=>200, msg=>, data=>（3层数据结构）

## 比赛列表

api: api/match/matchlist

post：match数据表字段、page(第x页) 一页10条数据

year（选择年份 有传入正确年份才查询完成时间在区间内的数据）；

keyword（搜索关键字，发布比赛的球队名）；

choice\_time（比赛时间选择 1：近7天|2：近15天|3：近1个月）

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 比赛列表（所有数据）

api: api/match/matchlistall

post：match数据表字段

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 比赛管理操作

api: api/match/removematch

post：match\_id（比赛id），action：editstatus（修改状态status字段）/del（删除比赛信息）

response:

成功 code=>200, msg=>'',

## 球队成员报名参加比赛

api: api/match/joinmatch

post: match\_id（match表id）

response:

成功 code=>200, msg=>'',

## 提交约赛应战信息

api: api/match/joinmatchapply

post:

match\_apply表字段：match\_id、match、team\_id（选择的team\_id）、team（选择的team）、telphone（提交的手机号）、contact（提交的联系人信息）

response:

成功 code=>200, msg=>'', data=>（新增match\_apply表id）

## 回复约赛应战

api:api/match/replymatchapply

post: apply\_id（match\_apply表id）、reply（2 同意|3拒绝）

response:

成功 code=>200, msg=>'', data=>（新增match\_apply表id）

## 球队参加比赛申请列表（页码）

api: api/match/matchapplypage

翻页传参: {:url('api/match/matchapplypage')}?page=页数

post: match\_apply数据表字段

response:

成功 code=>200, msg=>, data=>（3层数据结构）

## 球队参加比赛申请列表

api: api/match/matchapplylist

post：match\_apply数据表字段、page(第x页) 一页10条数据

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 球队战绩列表（有页码）

api: api/match/matchrecordlistpage

翻页传参 {:url('api/match /matchrecordlistpage)}/?page=页数

post: team\_id（球队id）、match表字段

year（选择年份 有传入正确年份才查询完成时间在区间内的数据）

response:

成功 code=>200, msg=>, data=>（3层数据结构）：match与match\_record表字段同一层结构

## 球队战绩列表

api: api/match/matchrecordlist

post：team\_id（球队id）、match表字段、page(第x页) 一页10条数据

year（选择年份 有传入正确年份才查询完成时间在区间内的数据）

response:

成功 code=>200, msg=>'', data=> （2层数据结构）：match与match\_record表字段同一层结构

## 球队战绩列表（所有数据）

api: api/match/matchrecordlistall

post：team\_id（球队id）、match表字段

response:

成功 code=>200, msg=>'', data=> （2层数据结构）：match与match\_record表字段同一层结构

## 历史对手球队列表（有页码）

api: api/match/historyteampage

翻页传参: {:url('api/match/historyteampage ')}?page=页数

post: team\_id（球队id）

keyword（搜索关键字，历史对手球队名）；

response:

成功 code=>200, msg=>, data=>（3层数据结构）

## 历史对手球队列表

api: api/match/historyteamlist

post: team\_id（球队id）、page(第x页) 一页10条数据

keyword（搜索关键字，历史对手球队名）；

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 历史对手球队列表（所有数据）

api: api/match/historyteamall

post：team\_id（球队id）

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 比赛战绩-会员关联列表（页码）

api: api/match/recordmemberpage

翻页传参: {:url('api/match/recordmemberpage')}?page=页数

post: match\_id （match表id）、status：1报名|2出席比赛|-1无效（剔除）

response:

成功 code=>200, msg=>, data=>（3层数据结构）

## 比赛战绩-会员关联列表

api: api/match/recordmemberlist

post： page(第x页) 一页10条数据、

match\_id （match表id）、status：1报名|2出席比赛|-1无效（剔除）

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 比赛战绩-会员关联列表（所有数据）

api: api/match/recordmemberall

post：match\_id （match表id）、status：1报名|2出席比赛|-1无效（剔除）

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

# 印象标签

## 印象标签列表（所有数据 无分页）

api: api/tag/tagall

post: (无)

response:

成功 code=>200, msg=>'', data=>（1层数据结构）

## 发表印象标签评论（一个标签只能评论一次）

api: api/tag/addtagcomment

post：

comment\_type 评论类型（1会员|2球队）

commented 被评论的实体名称（球队名之类）

commented\_id 被评论的实体id （球队id之类）

tag\_ids 提交的印象标签id集合，字符串，以逗号拼接（如1,2,3）

tags 提交的印象标签名集合，字符串，以逗号拼接（如球风良好,诚信守时）

response:

成功 code=>200, msg=>''

## 获取印象评论记录列表

api: api/tag/commentlist

post:

comment\_type 评论类型（1会员|2球队）

commented\_id 被评论的实体id （球队id之类）

response:

成功 code=>200,msg=>'', data=>（1层结构），其中data[].count是印象评论次数

# 球队

## 创建球队

api: api/team/createteam

post: 数据表字段对应

response:

成功:code => 200, msg=>'', insid=>id （insid为新增数据id）

## 编辑球队

api: api/team/updateteam

post: 数据表字段对应

response:

成功 code=>200, msg => ''

## 球队列表（有页码）

api: api/team/teamlistpage

翻页传参: {:url('api/team/teamlistpage')}/?page=x

post：（查询条件字段）

response:

成功:code=>200, msg=>'', data=>（3层数据结构）

## 球队列表（无页码）

api:api/team/teamlist

post：（查询条件字段） 、page(第x页) 一页10条数据

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 搜索球队列表（有页码）

api: api/team/searchteamlistpage

翻页传参: {:url('api/team/searchteamlistpage')}/?page=x

post：（查询条件字段）、搜索关键字keyword

response:

成功:code=>200, msg=>'', data=>（3层数据结构）

## 搜索球队列表

api: api/team/searchteamlist

post：（查询条件字段）、搜索关键字keyword 、page(第x页) 一页10条数据

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 我的球队列表

api: api/team/myteamlist

post:team\_member表字段，

response:

成功code=>200, msg=>'',data=>（2层数据结构）team详情信息在team{}中

## 球队成员列表

api: api/team/teammemberlist

post: team\_id

response:

成功 code=>200, msg=>'', data=>(2层数据结构),team详情信息在team{}中

## 移除球队成员（只有领队可操作）

api: api/team/removeteammember

post: team\_id（球队id）、member\_id（会员id）

response:

成功 code=>200, msg=>''

## 球队成员申请离队（成员自己操作）

api: api/team/applyleaveteam

post: team\_id（球队id）、member\_id（会员id）

response:

成功 code=>200, msg=>''

## 申请加入球队

api: api/team/applyjointeam

post: team\_id（球队id）、remarks（申请说明）

response:

成功 code=>200,msg=>'',insid=>xxx

## 回复申请加入球队结果

api: api/team/applyreply

post: apply\_id(apply表id)、reply(2 同意 3拒绝)

response:

成功 code=>200, msg=>'',

## 创建球队活动

api: api/team/createteamevent

post: 页面填写字段对应数据字段名

response:

成功 code=>200, msg=>'', data=>新增id(team\_event表id)

## 修改球队活动

api: api/team/updateteamevent

post: 页面填写字段对应数据字段名、参加活动人员(memberdata 格式形式与训练营活动一样)

response:

成功 code=>200, msg=>'',

## 直接创建并录入活动

api: api/team/directcreateteamevent

post: 页面填写字段对应数据字段名、参加活动人员(memberdata 格式形式与训练营活动一样)

response:

成功 code=>200, msg=>'', 新增id(team\_event表id)

## 球队活动列表（有页码）

api: api/team/teameventlistpage

翻页传参 {:url('api/team/teameventlistpage')}/?page=页数

post: 查询字段条件

year（选择年份 有传入正确年份才查询创建时间在区间内的数据）

response:

成功 code=>200, msg=>, data=>（3层数据结构）

## 球队活动列表

api: api/team/teameventlist

post：（查询条件字段） 、page(第x页) 一页10条数据

year（选择年份 有传入正确年份才查询创建时间在区间内的数据）

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 球队最新活动记录

api: api/team/lastevent

post：必须传入team\_id（球队id）、status=1（上架活动）、is\_finished=0（未完成）

response:

成功 code=>200, msg=>'', data=>（一维数组）

其中memberlist是活动人员名单列表，start\_time\_stamp是活动开始时间戳，is\_finished\_num是活动是否完成（0未完成|1已完成）

## 报名参加球队活动

api: api/team/jointeamevent

post: event\_id（球队活动id）

response:

成功 code=>200, msg=>'', data=>新增id(team\_event\_member表id)

## 球队活动-会员关联列表（有页码）

api: api/team/teameventmemberlistpage

翻页传参 {:url('api/team/teameventlistpage')}/?page=页数

post: 查询字段条件

response:

成功 code=>200, msg=>, data=>（3层数据结构）

## 球队活动-会员关联列表

api: api/team/teameventmemberlist

post：（查询条件字段） 、page(第x页) 一页10条数据

response:

成功 code=>200, msg=>'', data=> （2层数据结构）

## 球队活动-会员关联列表（无分页）

api: api/team/teameventmemberall

post:（查询条件字段）

response:

成功code=>200, msg=>'', data=>（2层数据结构）

## 球队活动、比赛评论列表（有页码）

api: api/team/teamcommentlistpage

翻页传参 {:url('api/team/teameventlistpage')}/?page=页数

post:

comment\_type 评论类型 1球队 | 2球队活动 | 3球队比赛

comment\_id 被评论实体id 根据评论类型所对应数据表的id字段

response:

成功 code=>200, msg=>, data=>（3层数据结构）, thumbup\_count=>（点赞数）

## 球队活动、比赛评论列表

api: api/team/teamcommentlist

post:

comment\_type 评论类型 1球队 | 2球队活动 | 3球队比赛

comment\_id 被评论实体id 根据评论类型所对应数据表的id字段

page(第x页) 一页10条数据

response:

成功 code=>200, msg=>, data=>（2层数据结构）, thumbup\_count=>（点赞数）

## 发布球队活动、比赛评论

api: api/team/addteamcomment

post:

comment\_type 评论类型 1球队 | 2球队活动 | 3球队比赛

comment\_id 被评论实体id 根据评论类型所对应数据表的id字段

comment 评论文字内容

response:

成功 code=>200, msg=>'', data=>新增id(team\_comment表id)

## 球队活动、比赛点赞/取消点赞

api： api/team/dianzan

post:

comment\_type 评论类型 1球队 | 2球队活动 | 3球队比赛

comment\_id 被评论实体id 根据评论类型所对应数据表的id字段

response:

成功 code=>200, msg=>'', thumbup\_count=>（最新的点赞数字统计）

## 获取球队活动、比赛点赞状态

api: api/team/isthumbup

post:

comment\_type 评论类型 1球队 | 2球队活动 | 3球队比赛

comment\_id 被评论实体id 根据评论类型所对应数据表的id字段

response:

成功 code=>200, msg=>'', thumbsup =>1（有点赞）|0（没点赞）

# 邀请|申请 Apply

## 获取招募|申请列表(page)

api: getApplyListByPageApi

post:

get:keyword

## 获取招募|申请列表(不带page)

api: getApplyListNoPageApi

post:

get:keyword

## 邀请加入|申请加入,写入一条

api: createApplyApi

post: {

"member": "woo",

"member\_id": 8,

"organization\_type": 1,

"organization": "准行者训练营",

"organization\_id": 4,

"type": 2,

"inviter": "legend",

"invite\_id": 6,

"remarks": "legen邀请woo加入教练",

"apply\_type":

}

PS: apply\_type:1:申请 2:邀请

## 邀请加入|申请加入,修改一条

api: createApplyApi

get: apply\_id

post: {

"member": "woo",

"member\_id": 8,

"organization\_type": 1,

"organization": "准行者训练营",

"organization\_id": 4,

"type": 2,

"inviter": "legend",

"invite\_id": 6,

"remarks": "legen邀请woo加入教练",

"apply\_type":

}

PS: apply\_type:1:申请 2:邀请

# 关注 Follow

## 添加关注/取消关注（暂缺球队）

api: api/follow/focus

post:

type: （1会员,2训练营,3班级,4球队）

member\_id/camp\_id/grade\_id/team\_id

## 我的关注列表（有页码）

api: api/follow/myfollow （分页api/follow/myfollow?page=xxx）

(无传参)

## 被关注（粉丝）列表（有页码）

api:api/follow/myfans （分页api/follow/myfans?page=xxx）

post:

type：（1会员,2训练营,3班级,4球队）

member\_id/camp\_id/grade\_id/team\_id

## 获取与当前关注实体（会员、训练营、班级、球队）的关注状态

api:api/follow/getfollow

post:

type：（1会员,2训练营,3班级,4球队）

member\_id/camp\_id/grade\_id/team\_id

response:

code=200 data=1/-1 （1代表正在关注，-1代表已取消关注或者未关注）

# 招募Recruitment

## 搜索招募

api: searchRecruitmentApi

get:keyword,order

post:

## 搜索招募(带page)

api: getRecruitmentListByPageApi

get:keyword,order

post:

## 获取招募列表

api: getRecruitmentListApi

post:

get:order

## 新增招募接口

api: updateRecruitmentApi

post:

PS:上传的地区字段为address

## 编辑招募接口

api: updateRecruitmentApi

get: recruitment\_id

post:

PS:上传的地区字段为address

## 参与招募

api: joinRecruitmentApi

get: recruitment\_id

post: linkman,contact,remarks, recruitment\_id, recruitment,[id]

如果第一次参与,id不要传,修改|更新参与内容和修改时间,请提交id

## 获取响应人名单(无分页)

api: getRecruitmentMemberListNoPageApi

post:

## 获取响应人名单(page)

api: getRecruitmentMemberListByPageApi

post:

# 订单Bill

## 获取会员订单接口

api:getBillListApi

post:略

## 编辑|添加订单

api:updateBillApi

get:bill\_id(编辑才传)

post:略

# 训练营Camp

## 搜索训练营

api: searchCampApi

get|post: keyword, province, city, area

## 获取训练营的列表有page

api: getCampListByPageApi

get|post: keyword, province, city, area

## 训练营列表

api: getCampListApi

post:略

## 更新训练营:

api:updateCampApi

post:{data}

get:'camp\_id'

## 新建训练营

api:createCampApi

post:{data}

## 获取用户创建的训练营id

api:isCreatCampApi

post/get:member\_id(不上传默认为当前用户)

return:训练营id

## 训练营评论

api: createCampCommentApi

## 获取评论列表

api: getCampCommentListByPage

post:略

## 获取热门课程（数据打乱）

api: getHotCampList

post:get: province, city

# 训练营关系camp\_member

## 申请成为训练营的某个身份

api: applyApi

get|post:'type','camp\_id','remarks'

ps:type: -1:粉丝|1:学生|2:教练|3:管理员|4:创建者

## 训练营人员审核

api: ApproveApplyApi

get|post:'id','status'

ps:status只接受1和-1

## 训练营人员变更

api: modifyApi

get|post:'id','type'

ps:type只接受-1,2,3,

-1:粉丝|1:学生|2:教练|3:管理员|4:创建者

## 查询身份

api: getCampMemberApi

get|post:'camp\_id'

## 获取有教练身份的训练营员工

api: getCoachListApi

get|post:camp\_id

2017/09/27： 新增返回data数据campmemberid字段

## 自定义获取训练营身份列表

api: getCampMemberListApi

get:page,keyword

post:camp\_id,type

## 自定义获取训练营身份列表分页（有页码）

api: getCampMemberListByPageApi

get: keyword

post:camp\_id,type

ps:type查询条件为= ;

例:type=2,则查询教练身份列表;

## 自定义获取训练营身份列表分页（无页码无分页）

api:getCampMemberListNoPageApi

get: keyword,type

post:camp\_id,type

PS:查询的type条件为>=;

例:type=2,则查询教练以及教练以上的身份列表;

## 关注训练营成为粉丝

api: focusApi

get:post:camp\_id,remarks

## 解除训练营-会员关系（辞退训练营人员）

api:{:url('api/campMember/removerelationship')}

post:campmemberid

response:

成功 code 100

# 教练Coach

## 搜索教练列表

api:searchCoachListApi

get/post:keyword

## 申请成为教练

api:createCoach

post:{data}

## 获取教练列表

api:getCoachListApi;

post:{data}

## 获取教练身份相关的训练营

api: campListOfCaochApi

post|get:member\_id

## 完善教练信息

api: updateCoachApi

get: coach\_id

post:{data}

# 场地Court

## 获取场地列表

api: getCourtListApi

get:camp\_id

post:{data}

camp\_id不传值则查询系统库

## 分页获取场地（带分页、有页码）

api: getCourtListByPageApi

post:

get:keyword

## 搜索场地

api: searchCourtApi

get:keyword,page

post:只能传court数据表存在的字段

## 更新场地

api:updateCourtApi

get:id

post:{data}

## 创建场地

api:createCourtApi

post:{data}

## 获取训练营下的场地库

api: getCourtListOfCampApi

post: 只能传court数据表存在的字段

## 把场地添加到自己的库

api:ownCourtApi

get|post:court\_id,camp\_id

## 不分页获取场地列表

api: getCourtListNoPageApi

post:get:camp\_id

## 获取courtcamp多条记录(带page)

api: getCourtCampListOfPageApi

post:

## 获取courtcamp记录(无page无分页)

api: getCourtCampListNoPageApi

post:status,court\_id

# 会员Member

## 完善会员资料

api:updateMemberApi

get:member\_id

post:{data}

## 提现申请

api: withdrawApi

post:balance 略

## 编辑个人资料

api: updateMemberApi

post:略

## 注销登陆

api: logout

# 登陆|注册Login

## 会员注册

api: http://hotpaas.com/api/login/registerApi

post:member,password,repassword,telephone

## 字段是否被占用

api:http://hotpaas.com/api/login/isFieldRegisterApi

post:field,value

return:1=被占用,0=未被占有

## 会员登陆

api: http://hotpaas.com/api/login/loginApi

post:username,password

username为telephone或者member

return:略

# 学生student

## 获取会员学生身份列表

api:getStudentsListApi

get:member\_id(不传默认为当前用户)

## 新增学生身份

api: createStudentApi

post:

## 更新学生

api:updateStudentApi

get:student\_id

post:略

# 证件Cert

## 获取证件列表

api: CertListApi

## 添加证件

api: createCertApi

post:略

## 修改证件

api: updateCertApi

get:cert\_id

post:略

# 教案Plan

## 获取教案列表(翻页)

api: planListApi

post:略

## 获取教案列表(不分页)

api: planListNoPageApi

post:略

## 获取教案列表(带页码)

api: getPlanListByPageApi

post:略

## 新建教案

api:createPlan

post:

PS: 训练项目传一个二维数组转str到接口,字段为exercise\_str 结构为

## 修改教案

api:updatePlan

post:

# 训练项目 Exercise

## 获取列表(默认无分页)

api: getExerciseListApi

post:

## 获取列表(带分页)

api: getExerciseListByPageApi

post:

## 编辑项目

api: updateExerciseApi

post:

get:exercise\_id

## 新增项目

api: updateExerciseApi

post:

# 课程Lesson

## 新增课程

api: updateLessonApi

post:略

PS:助教的变量名为:assistants,而不是assistant,同理,助教id的变量名为assistant\_ids;都用逗号隔开

## 编辑课程

api:updateLessonApi

get:lesson\_id

post:略

PS:助教的变量名为:assistants,而不是assistant,同理,助教id的变量名为assistant\_ids;都用逗号隔开

## 获取课程

api: getLessonListApi

post:略

PS:post数据里面需要提交status字段:0:未审核;1:正常(上线);-1:下架(下线)

## 获取购买了课程但是没有分配班级的学生

api: getActiveLessonStudentsApi

get:lesson\_id

## 获取毕业学生

api: getEduatedStudentsApi

get:lesson\_id

## 获取购买了课程也分配了班级的学生

api: getInactiveStudentsApi

get:lesson\_id

## 获取购买了课程的没毕业的学生

api:getStudentListOfLessonApi

get:lesson\_id

post:其他限制条件,比如增加一个班级条件grade\_id

## 审核课程:

api: checkLessonApi

get|post:camp\_id,lesson\_id,status

## 获取热门课程

api: getHotLessonListApi

post|get: province, city

# 课程-学生 lesson\_member

## 搜索课程学生带page

api: searchLessonMemberByPageApi

post:

PS:{rest\_schedule:3}则限定剩余课时小于3的学生

## 搜索课程学生不带page

api: searchLessonMemberNoPageApi

post:

## 获取课程学生数据带page

api: getLessonMemberListByPageApi

post:

get:keyword

PS:{rest\_schedule:3}则限定剩余课时小于3的学生

## 获取训练营下的学生带page(唯一)学生|体验生

api: getLessonMemberListOfCampByPageApi

post:

get:keyword

PS:{rest\_schedule:3}则限定剩余课时小于3的学生

## 获取与课程|训练营相关的学生|体验生-不带page

api: getLessonMemberListNoPageApi

post:

get:keyword

PS:{rest\_schedule:3}则限定剩余课时小于3的学生

## 获取训练营相同价格的课程带分页page带搜索

api: getSamePriceLessonMemberListByPageApi

post|get: camp\_id,cost,[ keyword, rest\_schedule]

## 获取未分配班级的学生列表-带page

api: getNoGradeMmeberListByPageApi

post:

# 活动Event

## 新增活动

api: updateEventApi

post:略

PS:省市区的变量为address,一个空格隔开

## 编辑活动

api:updateEventApi

get: Event\_id

post:略

PS:省市区的变量为address,一个空格隔开

## 搜索活动列表

api: searchEventApi

post:

get:keyword,order

## 获取活动列表

api: geEventListApi

post:略

get|post:keyword,order

## 获取班级活动接口

api: getEventListOfGradeApi

get:order

post:

PS: -1:离营|0:待审核 |2:退出|3:被开除|4:毕业 的学生不返回结果

## 获取活动列表Page

api: geEventListByPageApi

post:略

get|post:keyword,order

## 审核活动:

api: checkEventApi

get|post:camp\_id,Event\_id,status

## 录入活动

api: recordEventApi

post: [跟编辑班级一样]

PS: 活动时间的变量为event\_times,格式为2017-10-25 17:27或者2017年10月25日17:27或者2017/10/25 17:25

PPS: post里必须传event字段,且event不能更改;

PPPS:

post的data示例:

{

event:'xxxxx爬山活动',

address:'广东省 深圳市 南山区',

location: 'xxxx荔枝山',

memberData:[{member:张三,member\_id:5,id:4,status:3},{member:李四,member\_id:2,id:88, status:3}],

}

pppps:如果是中途请假或者退出或者其他情况,status为4

## 参加|购买活动()

api: joinEventApi

get:event\_id,total

post: [memberData]

[memberData]示例:

[

['key'=>值,'is\_pay'=>0],

['key'=>值, 'is\_pay'=>1],

];

# 活动人员event\_member

## 获取人员列表(唯一)

api: getEventMemberListOfCampByPage

get:keyword

post:

## 获取与活动相关的会员(page)

api: getEventMemberListByPageApi

get:keyword

post:

## 获取与活动相关的会员(无page)

api:getEventMemberListNoPageApi

get:keyword

post:

# 班级Grade

## 注意事项:

由于部分数据需要特殊处理,所以,以下数据的提交字段名请变化一下(在前端输出其实也变了的)

assistant -> assistants

## 创建班级

url:createGradeApi

post: [gradeData],[ studentList]

[studentList]示例:

[

['key'=>值],

['key'=>值],

];

PS:示例为PHP的数组,不是JS的

PS:助教的变量名为:assistants,而不是assistant,同理,助教id的变量名为assistant\_ids;都用逗号隔开

## 获取班级列表

api: getGradeListApi

post:略

## 获取班级列表（不分页）

api: getGradeListAll

post: grade表字段；keyword（关键字 搜索班级名）；

默认查询status=1的数据

## 更新班级

url:updateGradeApi

get:grade\_id

post:略

PS:助教的变量名为:assistants,而不是assistant,同理,助教id的变量名为assistant\_ids;都用逗号隔开

## 班级单个学生变动(编辑班级可用)

api: updateGradeMemberApi

get:id

post:['grade\_id','grade']

PS:示例为PHP的数组,不是JS的

## 班级批量学生变动(编辑班级可用)

api: updateGradeMemberAllApi

post:[data]

PS:[data]必须包含id,例如

变动学生示例:

[

{student: "荣", student\_id: "51", id: "64", grade: "13213", grade\_id: "26"}

{student: "woo", student\_id: "54", id: "61", grade: "13213", grade\_id: "26"}

];

增加学生示例:

[

{student: "荣", student\_id: "51", grade: "13213", grade\_id: "26"}

{student: "woo", student\_id: "54", grade: "13213", grade\_id: "26"}

];

删减学生示例:

[

{student: "荣", student\_id: "51", id: "64", grade: "0", grade\_id: "0"}

{student: "woo", student\_id: "54", id: "61", grade: "", grade\_id: "0"}

];

PS:示例为PHP的数组,不是JS的

## 获取教练下的班级

api: getGradeListOfCoachByPageApi

get:coach\_id

# 班级-学生关系 grade\_member

## 获取数据列表带page

api: getGradeMemberListByPageApi

post:

get:keyword

## 获取与课程|班级(已分班)|训练营相关的学生|体验生-不带page

api: getGradeMemberListNoPageApi

post:

get:keyword

# 课时Schedule

## 判断课时冲突(2个小时内只允许一条数据)

api: recordScheduleClashApi

get|post:' lesson\_id' ,' lesson\_time', ' grade\_id',' camp\_id'

data:0|1;

ps:lesson\_time格式为2017-10-12 19:13:43

PS:返回1的时候,存在冲突,返回0的时候,不存在冲突

## 判断是否有录课权限

api: recordSchedulePowerApi

get|post:camp\_id

data:返回权限数字编号

## 课时审核

api: recordScheduleCheckApi

get|post:camp\_id

## 录课

api: recordScheduleApi

post:略

PS:上课时间的变量为: lesson\_time\_date,数据格式为:2017-10-12 19:05:02

PS:助教的变量名为:assistants,而不是assistant,同理,助教id的变量名为assistant\_ids;都用逗号隔开

## 课时评分

api: starScheduleApi

post:略

## 获取列表有page

api: getScheduleListByPageApi

post:略

## 获取开始时间和结束时间的列表带page

api: getScheduleListBetweenTimeByPageApi

get: begin, end,camp\_id

post:略(camp\_id必带);

PS:begin和end的参数格式为:

"2009-10-21 16:00:10"

"10 September 2008"

"+1 day"

"+1 month"

"+1 year"

"-1 day"

'2017-10-12"

......

# 课时会员关系 schedule\_member

## 获取训练营下的课程(课时名称唯一)

api: getScheduleMemberListOfCampByPageApi

post:

get:keyword

## 获取课时列表(page)

api: getScheduleMemberListByPageApi

post:

get:keyword(搜索的是user相关的课时)

## 获取课时,无分页

api: getScheduleMemberListNoPageApi

post:

get:keyword

# 消息Message

## 获取全部个人消息和系统消息列表

api: getMessageListApi

get|post:status

data: ['messagelist'=>$messageList,'messageMemberList'=>$messageMemberList]

status不传默认获取全部状态消息,系统消息不存在已读|未读状态;

## 获取系统消息详情

api: getMessageInfoApi

get|post: message\_id

## 获取个人消息详情

api: getMessageInfoApi

get|post: message\_id

## 设置消息状态

api: setMessageMemberStatus

get|post: message\_id, status

## 删除个人消息

api: removeMessageMemberApi

get|post: message\_id

## 获取个人消息列表

api: getUnReadMessageMmeberListApi

post:

# 一些辅助功能

## 获取用户地址Location

api: getLocation

get:ip(非必须)

## 获取用户IPLocation

api:getIP

## 获取??