定向越野ING文档

微信小程序大赛

目录

[目录 1](#_Toc15523)

[1. 成员与分工 5](#_Toc23645)

[2. 项目介绍 6](#_Toc7277)

[2.1 项目简介 6](#_Toc886)

[2.2 应用背景和目标用户 6](#_Toc2524)

[2.3 选题意义 6](#_Toc28825)

[3. 需求分析 7](#_Toc2373)

[3.1 总体描述 7](#_Toc26759)

[3.1.1 系统背景 7](#_Toc24351)

[3.1.2 业务需求 7](#_Toc19889)

[3.1.3 用户特征 7](#_Toc7328)

[3.1.4 假设和依赖 8](#_Toc27343)

[3.2 功能需求 8](#_Toc28280)

[3.3 非功能需求 8](#_Toc31564)

[3.3.1 安全性 8](#_Toc9750)

[3.3.2 可维护性 8](#_Toc10202)

[3.3.3 易用性 8](#_Toc31318)

[3.3.4 可靠性 8](#_Toc8434)

[3.3.5 性能需求 9](#_Toc8862)

[3.4 数据需求 9](#_Toc3597)

[3.4.1 数据定义 9](#_Toc18331)

[3.4.2 默认数据 9](#_Toc32719)

[3.4.3 数据格式要求 9](#_Toc24393)

[4 产品设计 9](#_Toc17545)

[4.1 用户界面分解 9](#_Toc30192)

[4.2 UI设计 10](#_Toc14772)

[4.2.1 标准色 10](#_Toc13360)

[4.2.2 字体 13](#_Toc19027)

[4.2.3 控件 13](#_Toc12891)

[4.3 使用流程 13](#_Toc2399)

[4.3.1 创建活动 13](#_Toc27562)

[4.3.2 邀请 14](#_Toc4273)

[4.3.3 打卡设置 14](#_Toc30739)

[4.3.4 编辑管理员 15](#_Toc20476)

[4.3.5 组建小队（成为队长） 16](#_Toc10691)

[4.3.6 编辑队员 16](#_Toc5335)

[4.3.7 查看成绩 16](#_Toc21395)

[4.3.8 分享成就 17](#_Toc7148)

[5. 技术分析 17](#_Toc26602)

[5.1 架构设计 17](#_Toc15554)

[5.2 数据库设计 19](#_Toc23533)

[5.2.1 数据需求收集和分析 19](#_Toc3268)

[5.2.2 概念结构设计 19](#_Toc22762)

[5.2.3 逻辑结构设计 20](#_Toc31589)

[5.2.4 物理结构设计 20](#_Toc22113)

[5.2.5 数据库实施 20](#_Toc14489)

[5.2.6 运行和维护 20](#_Toc18716)

[5.3 功能模块划分 21](#_Toc3)

[5.3.1 活动管理 21](#_Toc14382)

[5.3.2 参与者管理 22](#_Toc30331)

[5.3.3 打卡管理 23](#_Toc20186)

[5.3.4 用户中心 25](#_Toc29916)

[5.4 实现方案 25](#_Toc31400)

[5.4.1 前端技术 25](#_Toc12863)

[5.4.2 后端技术 25](#_Toc24374)

[5.4.3 部分核心技术分解 26](#_Toc4363)

[5.5 测试 28](#_Toc30091)

[5.5.1 功能测试 28](#_Toc11650)

[5.5.2 性能测试 30](#_Toc6892)

[6. 应用运营 30](#_Toc431)

[6.1 上线发布 30](#_Toc13199)

[6.2 运营维护方案 30](#_Toc20970)

[6.2.1 引流策略 30](#_Toc26518)

[6.2.2 存留策略 30](#_Toc28233)

[6.2.3 优化策略 30](#_Toc30012)

[6.3 实际运行数据 31](#_Toc30895)

[7. 总结 35](#_Toc10527)

# 1. 成员与分工

|  |  |
| --- | --- |
| 姓名 | 分工 |
| 倪奕晨（组长） | 参与者管理模块、UI设计 |
| 邓楚宸 | 首页、用户中心模块、美工 |
| 何青云 | 活动管理模块 |
| 徐王子 | 打卡管理模块 |

# 2. 项目介绍

## 2.1 项目简介

“定向越野ING”是一款可以自主发起、制定和参与定向越野活动的微信小程序。活动分为个人赛与团体赛，用户可以自行选择活动的时间、地点，并且自行制定定向越野路线。小程序将根据所设打卡点个数生成相应二维码，当用户通过手机扫描二维码完成全部地点的打卡后，小程序将记录成绩并进行排名。活动结束后用户还可以查看活动成绩单、成就等。

## 2.2 应用背景和目标用户

目前通过“定向越野”等关键词可以搜索到的微信小程序中，各个产品的质量参差不齐，存在诸多问题：无法定制活动路线和打卡方式、无法很好地收集参赛人员打卡过程的信息、实际使用不稳定、胡乱收费等。

由于疫情原因，南京大学定向越野队的校外户外训练不得不改为校内进行。由于没有校外专业设备的支持，定向越野队队员的日常训练数据无法有效地被收集和保存。

“定向越野ING”这款微信小程序使用了微信提供的云开发功能，这使得无论是活动的发起方还是参与者都可以十分便利地开展工作——完成发起活动、比赛打卡、统计成绩等一系列操作。随着4G、5G技术的普及，依托移动互联网实现实时信息传输，“定向越野ING”与传统定向越野比赛使用的电子打卡装置相比，可以在实现原有所有功能的基础上，实现流程化和实时信息同步，可以更加高效地记录打卡数据和统计成绩；同时，使用“定向越野ING”不必租用专业设备，降低了开展定向越野活动的门槛。

“定向越野ING”目前面向的用户主要为接触定向越野的大学生以及年轻人，但随着定向越野的普及与流行，其受众将会越来越广泛，适用年龄段也将拓宽。目前我们团队已经与南京大学定向越野队达成合作，该小程序已经投入到了校定向越野队的日常练习与比赛中，帮助南京大学定向越野队更加顺利地开展日常训练、减轻校队管理者的工作负担。

## 2.3 选题意义

新冠疫情发生以来，人们的生活方式发生巨大改变，普遍疏于锻炼，身体素质有所下降，而且在心理素质方面也存在严重问题。而积极参加体育锻炼能提高免疫力，增强抗病毒能力，更能缓解压力，消除疫情产生的恐慌心理。2020 年 9 月国家体育总局和教育部联合发布的《关于深化体教融合促进青少年健康发展的意见》，更是对青少年的健康人生和中国体育的可持续发展指明方向。因此对于体育运动、体质提升的关注是很有必要的。而后疫情时代大学生对于健身APP的选择从疫情前的15%上升到了31%，可见健身小程序的发展是极具前景的。

但近年来，由于传统体育的特点没有跟上时代的发展，大学生参加体育锻炼的积极性降低。因此，定向越野作为新型的运动方式引入大学可以激发大学生有意识地参与体育锻炼的动机。定向越野在大学生日常生活中可以体现很重要的作用，尤其是对大学生的独立性有很大的帮助。学生们在参加定向活动过程中享受着团队合作，较量着智力，学生在不断地判断地形和选择路线以及成功到达目标点获得成就感，不断地改善自己的身体素质、耐力以及最重要的意志力。

同时，由于疫情原因，在校大学生的出行受到限制，许多运动类的校队和社团举办活动受阻，导致许多原本能够为在校大学生提供日常锻炼的训练和比赛无法开展。南京大学定向越野队便是诸多受到疫情影响的校运动队之一。

综上所述，在后疫情时期，为提高广大民众身体素质和心理健康，同时为定向越野爱好者开展活动提供便利，我们制作这款小程序以辅助定向越野活动的开展，用以满足大学生和广大民众的运动以及娱乐需求。

# 3. 需求分析

## 3.1 总体描述

### 3.1.1 系统背景

传统的定向越野活动，准备时间长，准备流程复杂，参赛者的组织较困难，成绩统计周期长。微信小程序“定向悦野ING”就是为改变这一现象，鼓励同学加入到定向越野活动中而开发的。活动主办方可以通过本系统实现从创建活动、准备活动，到进行活动，再到赛后成绩统计的全过程，为激发同学参与热情，又设计了成就模块。

### 3.1.2 业务需求

系统使用后，创建者和参与者可以进行一系列定向越野活动，也可查看已结束的活动成绩和个人特殊成就。

### 3.1.3 用户特征

活动创建者（管理员）：每位活动有一位创建者，若干位管理员，目标是举办小到中型（10~100人）的定向越野活动，希望通过系统生成打卡点密码/二维码，并可自动设置打卡路线，由系统根据路线进行打卡成功与否的逻辑判定。他们并不参加活动，但需要在活动中实时获知参与者打卡的时间信息，并在赛后能得到每位参赛者的最终成绩。

参与者：目标是参与已在本系统中创建的活动。希望获得公平的打卡成绩记录，并能在赛后获知自己的成绩。

### 3.1.4 假设和依赖

AE1：每名用户参与的活动数量不大于100。

AE2：管理员不同时参加活动（管理员与参赛者身份互斥）。

## 3.2 功能需求

1.主办方发活动。

2.参赛者加入活动。

3.参赛者（队长）创建团队（团队赛中）。

4.参赛者（队长）向团队中添加队员（团队赛中）。

5.根据主办方需要，自定义活动具体信息（打卡方式、赛制、GPS辅助定位、路线）。

6.在活动进行中，根据活动规则，记录并判定参赛者打卡用时。

7.在活动进行中，为活动管理人员显示活动中的参赛者打卡时间信息。

8.活动结束后，参赛者可以查看个人的打卡时间信息。

9.可以进行参赛者“特殊成就”的判定。

## 3.3 非功能需求

### 3.3.1 安全性

Safety1：管理员只能查看、无法删除活动打卡记录。

Safety2：参赛者只能查看自己的打卡信息。

### 3.3.2 可维护性

Modifiability1：若要增加新的赛制，4人要能在两周内完成。

Modifiability2：若要增加新的成就，1人要能在一天内完成。

### 3.3.3 易用性

Usability1：扫描二维码、输入密码的按钮要醒目，能在移动中看清。

### 3.3.4 可靠性

Reliablility1：若参赛者在活动中退出界面，系统要能保存此前的打卡记录。

### 3.3.5 性能需求

PR1：

（最低标准）在200个用户并发时，系统80%的请求不能超过100ms。

（一般标准）在200个用户并发时，系统90%的请求不能超过100ms。

（一般标准）在200个用户并发时，系统99%的请求不能超过100ms。

PR2：系统至少能储存1000名用户的信息

PR3：系统至少能为每个用户储存100次活动信息

PR4：98%的查询请求不能超过100ms

## 3.4 数据需求

### 3.4.1 数据定义

系统需要存储的数据实体及其关系参见5.2数据库设计模块。

### 3.4.2 默认数据

默认数据用于向系统中增加数据时

Default1：活动日期默认为当天日期。

Default2：参加活动后，用户身份默认为参赛者。

Default3：团队赛中，参赛用户身份默认为未参加小队者。

### 3.4.3 数据格式要求

Format1：路线各点以半角逗号，即“,”隔开。

Format2：活动ID为非负整型数。

Format3：活动密码为四位数字。

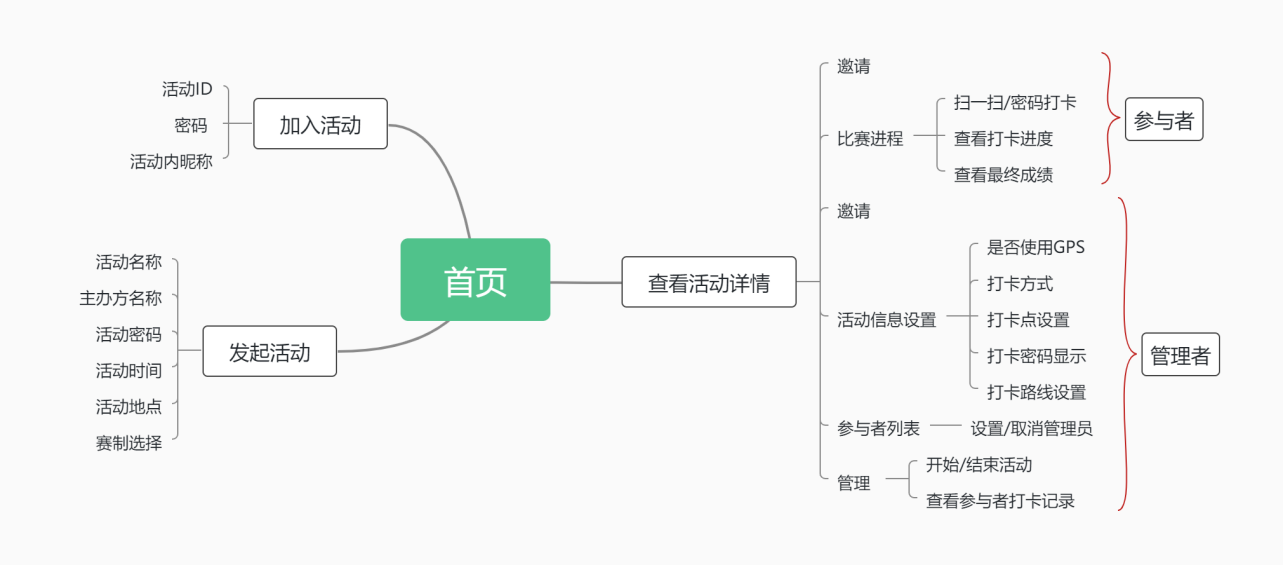
Format4：活动名称、主办方名称、活动内参赛者昵称均为字符串（不含空格）。

# 4 产品设计

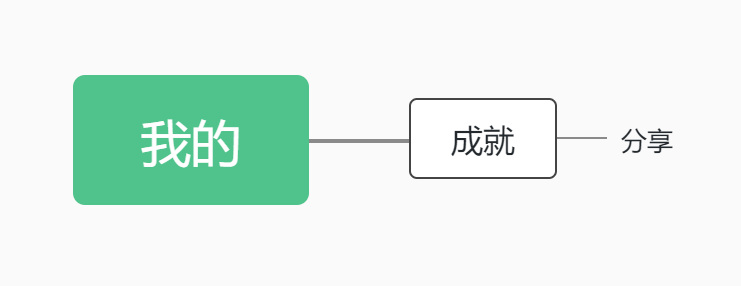
## 4.1 用户界面分解

为便捷用户使用，我们将用户界面分解为首页和“我的”两大板块，避免了页面切换过多导致体验感降低。

在首页，用户可以选择加入活动、发起活动、点击具体活动查看活动详情，也可以通过下方的导航栏去往“我的”页面。



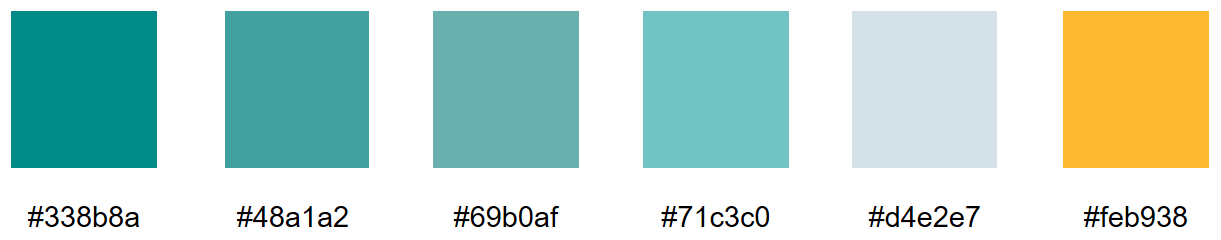
在“我的”页面，用户可以查看自身成就并进行分享。



## 4.2 UI设计

### 4.2.1 标准色

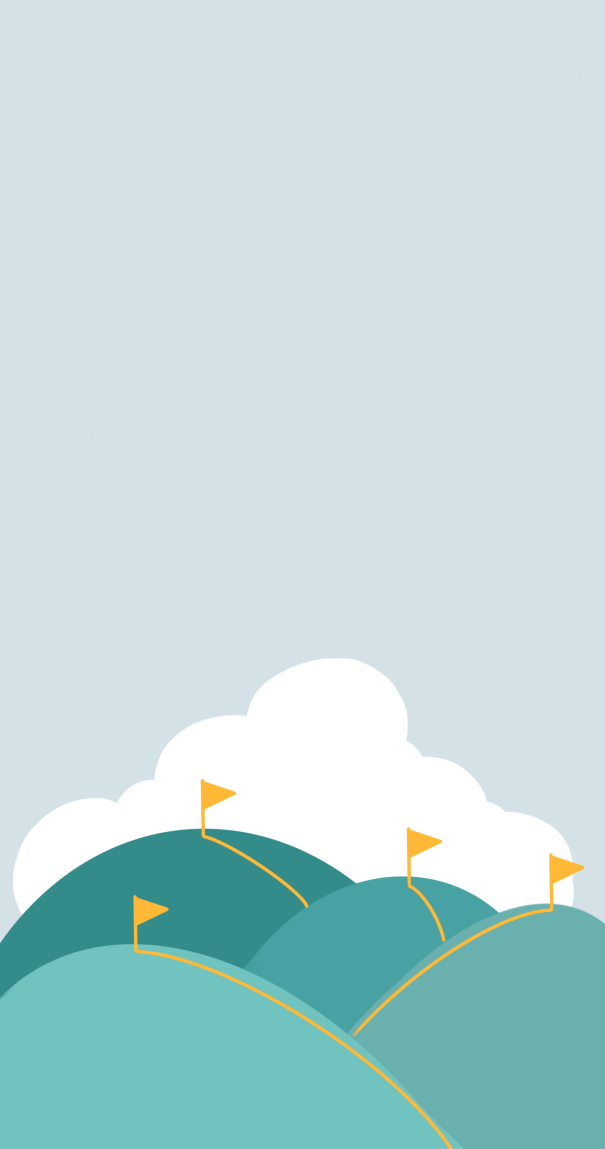
由于小程序定位为运动健身，我们选用了绿色和白色作为主色，清新自然，简洁大方。为了规范，我们统一了各个控件与图标的绿色色号。另外，我们还使用了少量的黄色，例如：标签页的激活提示，这样一来，搭配和谐的同时也活力明亮。



实际上，产品用色也与小程序的图标相统一，显得协调美观。



在处理背景颜色时，为提升小程序的美观程度，我们自行绘制了背景图片。整体上来看该图简洁漂亮，将绘制的内容集中在了界面下方，避免了遮挡界面中的组件，在丰富界面的同时也不喧宾夺主。



### 4.2.2 字体

引入多种字体，在多种不同型号的手机上显示正常。

### 4.2.3 控件

在控件方面，我们统一使用了Vant Weapp这一UI组件库。而对于不同的实现需求，我们挑选了功能上和用户使用角度最合适、舒适的组件；对于功能相似的部分，我们使用同一组件，避免了组件不同而造成的混乱，且风格统一，视觉和谐。

例如：

对于管理员完善活动信息、参与者打卡，采用Steps步骤条，展示操作流程的各个环节，让用户了解当前的操作在整体流程中的位置。

在需要管理员确认开始活动与结束活动时，为防止用户误操作，我们采用了Dialog弹出框，要求用户进行再确认方可执行操作。

在众多场景中，我们采用了Toast轻提示，在页面中间弹出黑色半透明提示，用于消息通知、操作结果提示。

值得一提的是，我们在一众控件中都使用了圆角，呈现效果更加轻松温和，改善了眼睛的视觉流，使布局和元素更容易在视觉上遍历，提高可用性，并最终提高目标用户的参与度。除此之外，我们还在首页与管理界面加入了具有一定灰度的透明板，灵活适配背景的同时也增加了立体感。

## 4.3 使用流程

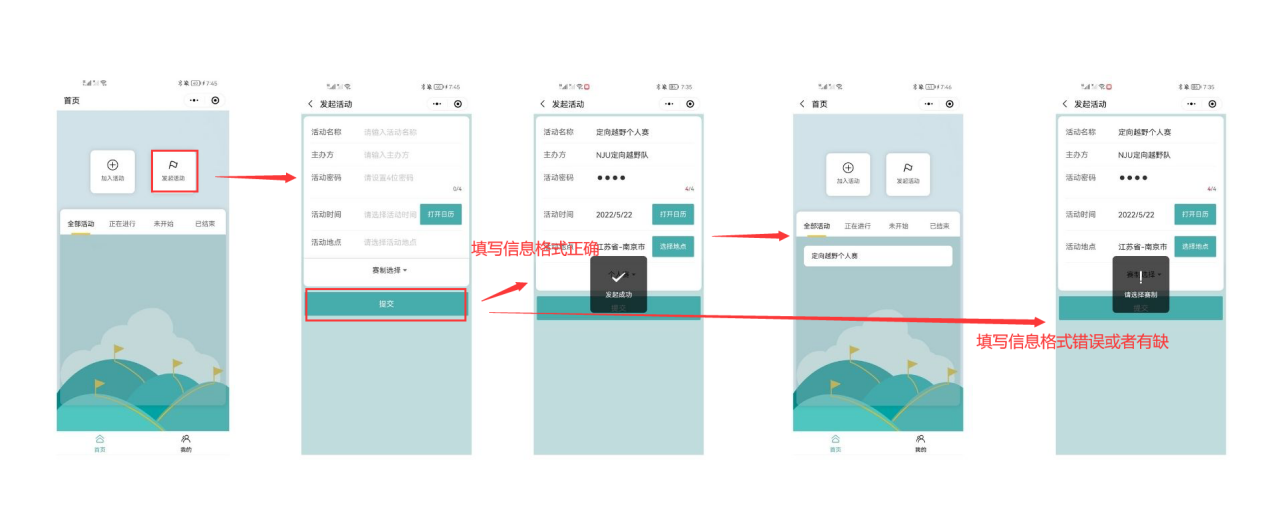
为加强流程操作的易懂性，让用户获得出色的体验，我们设置了逻辑清晰的使用流程。基于小程序轻快的特点，在充分尊重用户知情权和操作权的基础之上，建立起友好、高效、一致的用户体验。

为避免用户迷失，我们的每个页面都有明确的重点，以便于用户每进入一个新页面的时候都能快速地理解页面内容。我们还为首次使用的用户提供了用户指南，便于快速了解程序基本流程。

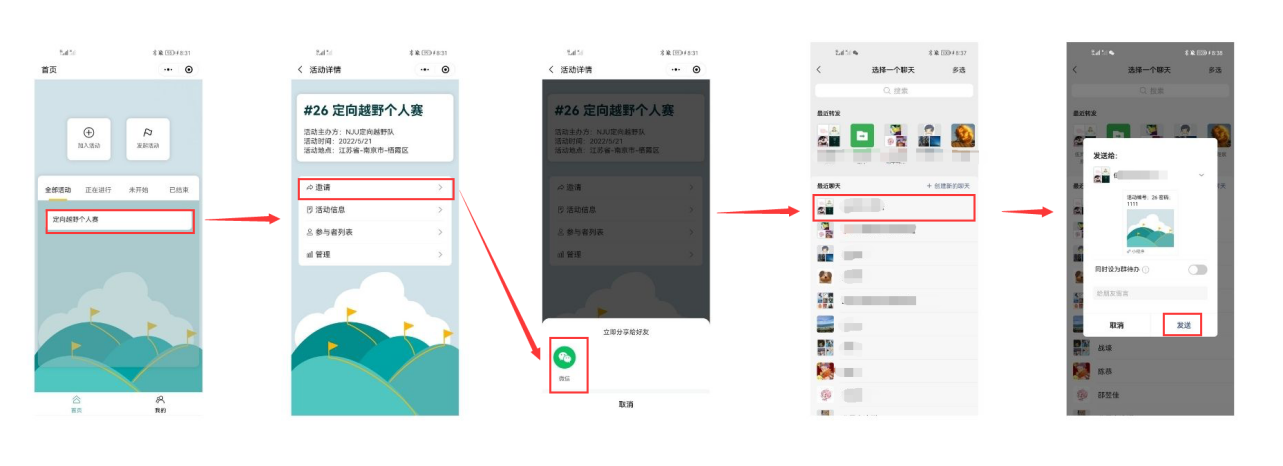
以下提供部分主要流程：

### 4.3.1 创建活动

管理员点击进入发起活动界面，填写相关信息时，为便捷用户输入，“打开日历”可弹出日历，“选择地点”可弹出省市二级联动。若填写信息缺失或不符合规范（密码长度不足）将弹出相应的错误提示，便于用户修改。当填写正确，将显示发起成功，自动返回首页，可在此看到新发起的活动，操作自然，流程清晰。



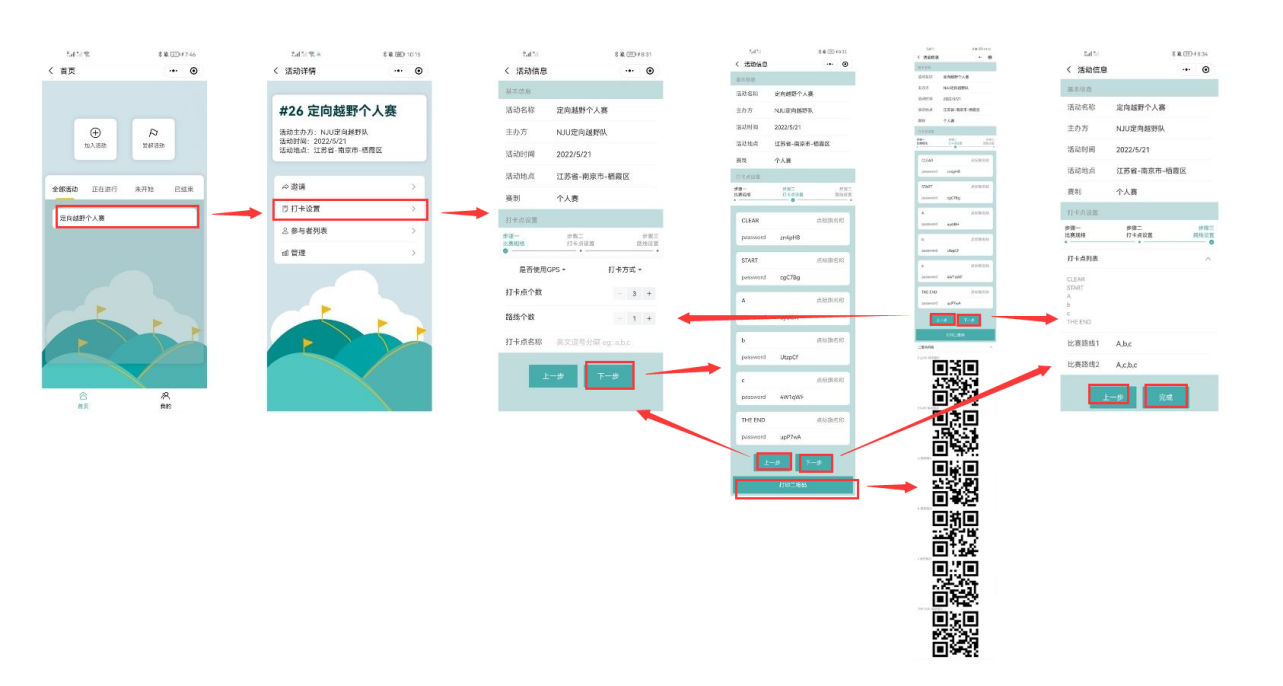
### 4.3.2 邀请



### 4.3.3 打卡设置

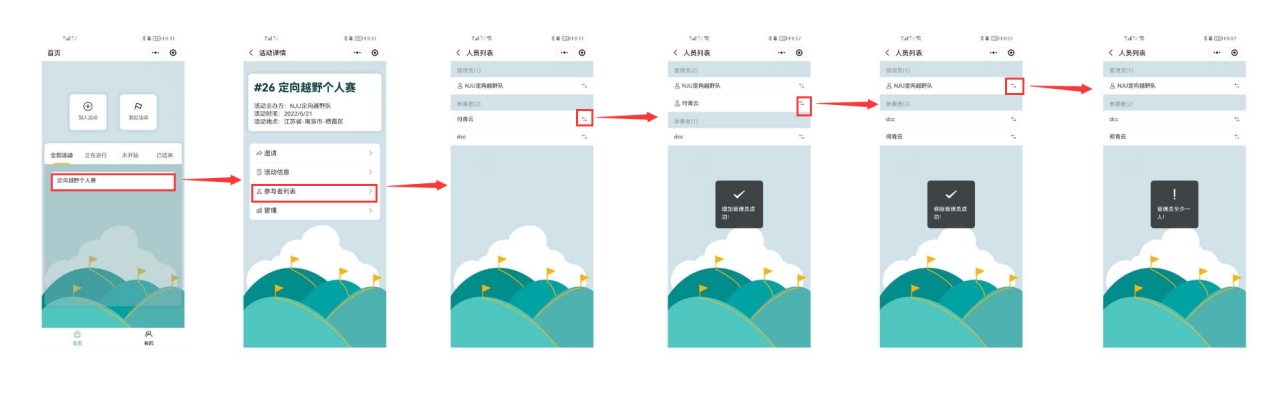
管理员在发起一项活动后，点击该项活动，基本信息栏显示此前填写的信息，打卡点设置栏若填写不完全将弹出相应的错误提示。若选择扫描二维码的打卡方式，用户可点击保存图片将二维码保存至本地相册。

在活动开始前，管理者可多次修改该页设置。

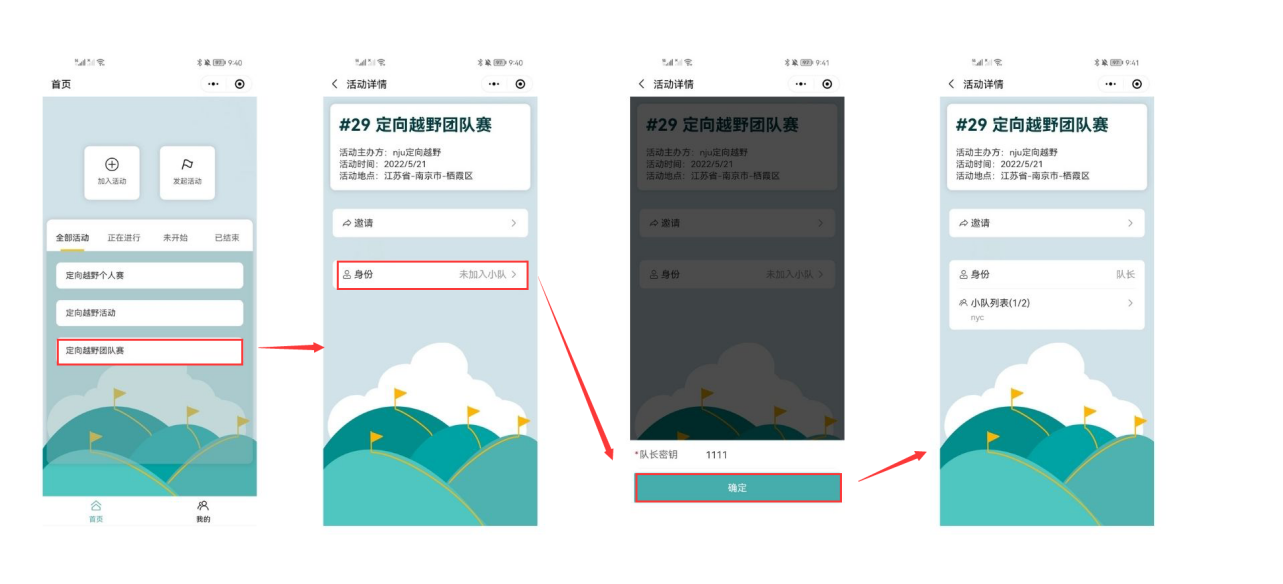


### 4.3.4 编辑管理员

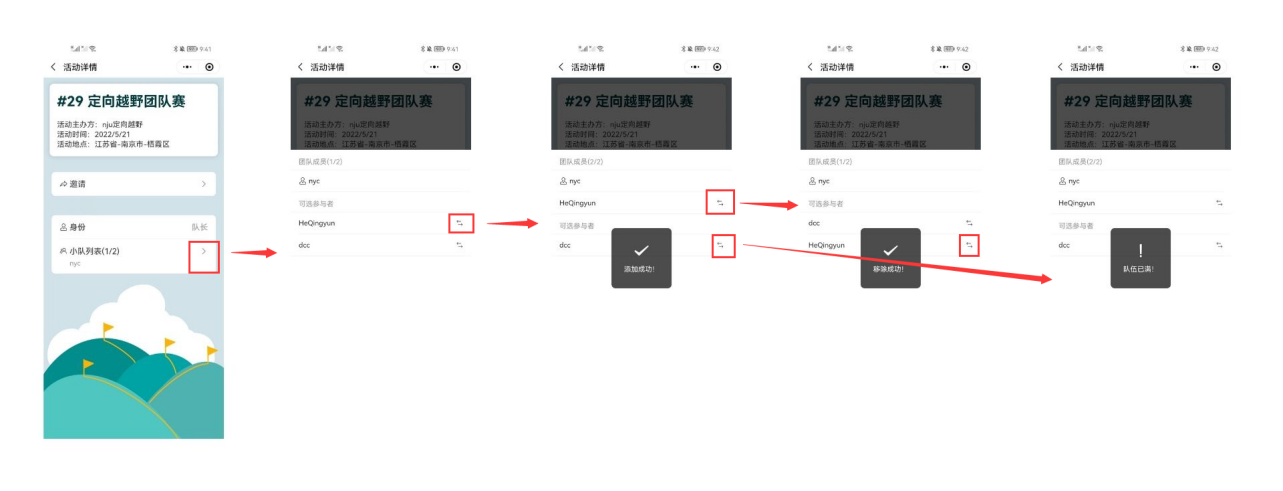
根据活动实际情况，管理员可将其他参与者设置为管理员，这一功能，十分贴合用户需求，为管理员与参与者的身份转换提供了解决方案。



### 4.3.5 组建小队（成为队长）

当加入的比赛为团队赛时，通过输入密钥可以成为队长，查看与编辑小队列表。

### 4.3.6 编辑队员



### 4.3.7 查看成绩

点击活动结束后，活动状态更改为已结束，在首页进行变更。结束活动后管理员仍然可以进入管理界面，符合现实需求，完善用户体验。



### 4.3.8 分享成就

解锁成就后可以查看相应内容并且分享给微信好友，进一步丰富了小程序的功能，使其更加成熟，激发了用户的探索欲，让定向越野这一活动更富有乐趣。

# 5. 技术分析

## 5.1 架构设计

本系统的架构采用分层结构，总共划分为六层：

1. 数据层为系统提供数据支持。本系统采用微信云开发数据库作为主要的数据库进行数据存储，稳定可靠，支持在小程序端和云函数中调用。
2. 支撑层为系统的设计开发提供支持。本系统利用微信云开发平台，不同于传统程序中前后端联调的模式，使得前端一站式解决。
3. 服务层包含了系统运行过程中所需的一些公共基础服务，如邀请、分享、用户登录等。
4. 业务层包含了系统主要的业务服务：活动管理、参与者管理、打卡管理、用户中心。
5. 逻辑层包含了系统的判断逻辑，负责对显示层传入的消息进行业务分派，并向视图层发送更改信息
6. 显示层负责页面显示，接受用户输入，向逻辑层传递消息，并且通过双向绑定实现实时渲染。



## 5.2 数据库设计

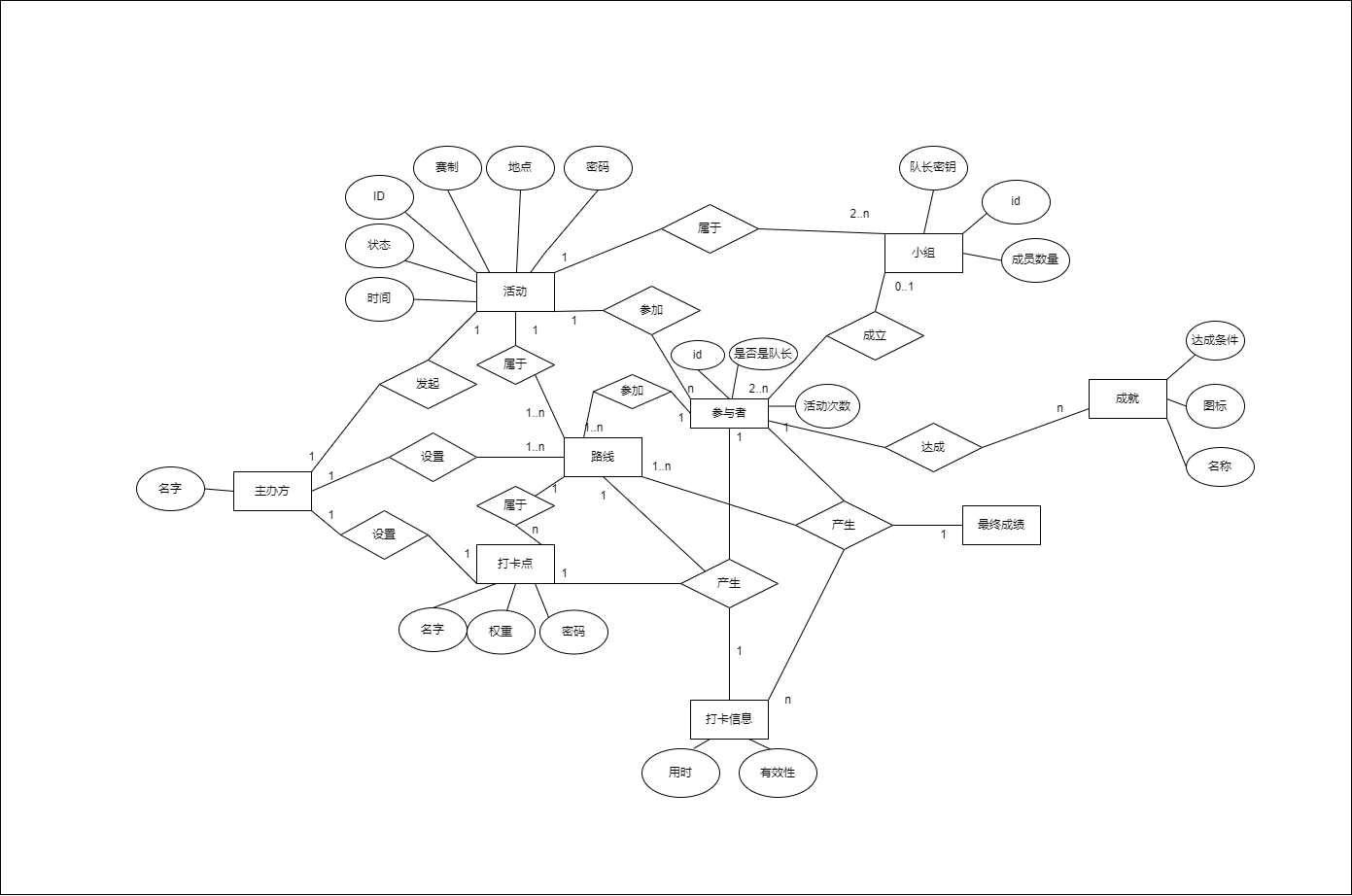
本系统数据库设计综合了基于ER模型的数据库设计方法与面向对象的数据库设计方法。

### 5.2.1 数据需求收集和分析

与南京大学定向越野队进行讨论，建立系统的数据字典。

### 5.2.2 概念结构设计

根据数据字典抽象出的概念模型ER图如下：



### 5.2.3 逻辑结构设计

根据E-R图，设计数据库。由七张表组成，分别是：activities, details, pointsTable, users, results, actors, groups。

### 5.2.4 物理结构设计

在云开发数据库本身提供的\_id作为索引之外，根据系统数据字典数据流的频度，为提升查询速度，又建立新索引如下：

表 activities 属性 *id* 的索引 *real\_id*

表 groups 属性 *\_openid* 的索引 *groups\_openid\_index*

表 pointsTable 属性 *\_openid* 的索引 *pointsTable\_openid\_index*

经调用次数统计，索引达到预期效果。

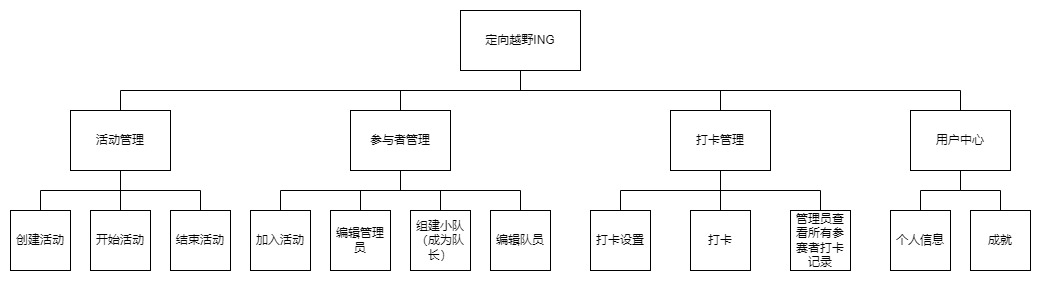
### 5.2.5 数据库实施

数据库已建立。

### 5.2.6 运行和维护

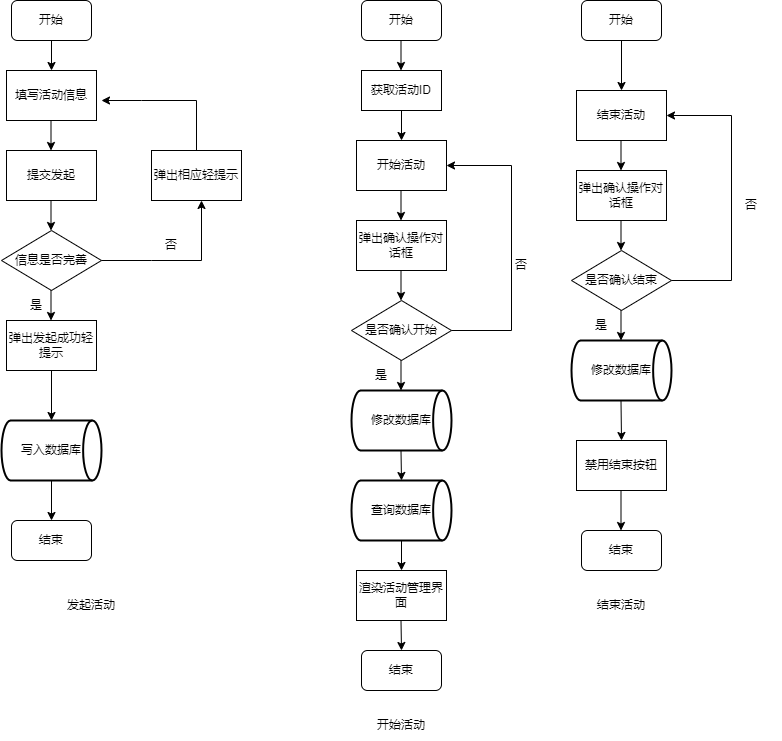
此后将定期（每星期）对数据库进行检查和维护。

## 5.3 功能模块划分



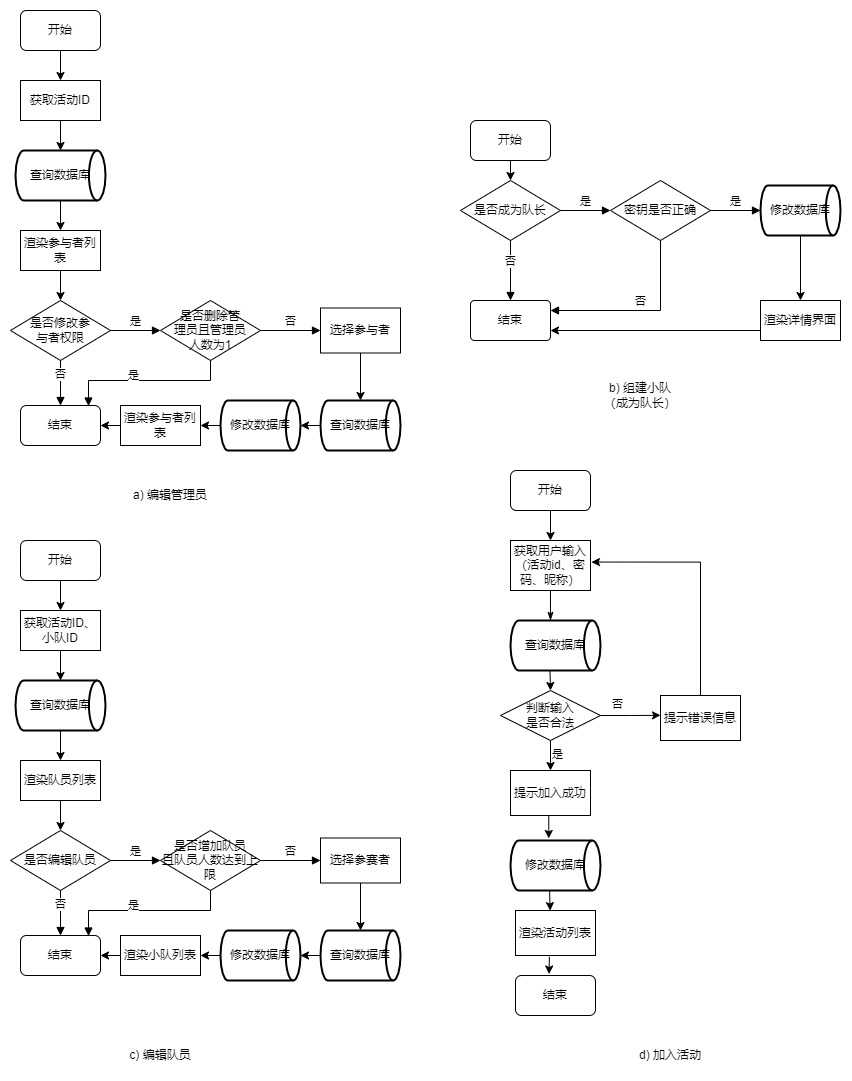
### 5.3.1 活动管理

活动成功发起后管理员可以选择开始活动，经过确认后该活动状态变为正在进行，管理页面显示各参与者的到达情况。管理员结束活动后经确认该活动状态变为已结束，管理员仍可查看参与者成绩。



### 5.3.2 参与者管理

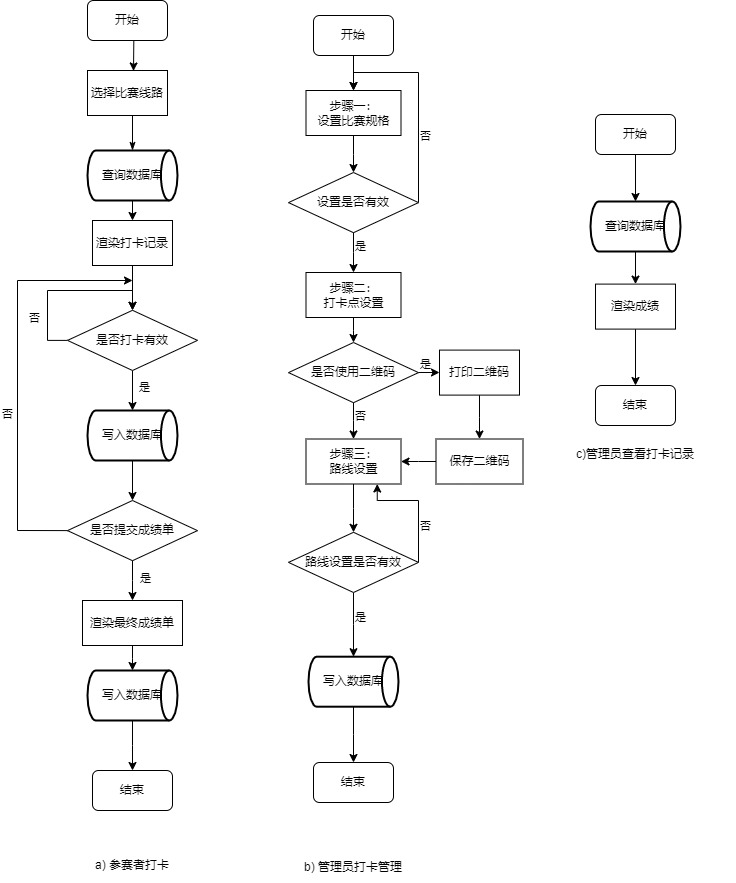
活动的参与者一级身份可以分为参赛者和管理员。管理员有权限增加和删除管理员，但每个活动限制至少有一个管理员。当活动类型为团队赛时，参赛者会有二级身份，分为：未加入活动者、队长和队员。未加入活动者可以成为队长。队长可以邀请未加入活动者成为其队员，也可删除队员。逻辑细节详见5.4.3.3。参与者管理模块帮助管理参与者身份，而不同的身份页面显示和提供的功能各不相同。



### 5.3.3 打卡管理

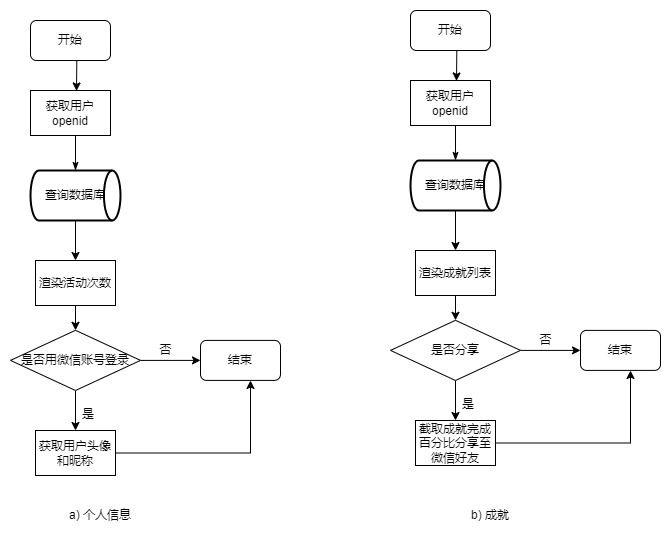
参赛者打卡时，打卡数据会实时写入数据库，确保参赛者不小心退出当前页面后仍可以恢复之前的打卡记录。参赛者完成打卡后，可以获得最终成绩单，并且管理者可以看到所有人的成绩排名。

管理员进行打卡管理时，打卡管理数据会实时写入数据库，确保管理员不小心退出当前页面后仍可以恢复之前的设置数据。



### 5.3.4 用户中心

用户中心的“个人信息”中展示用户参加的活动次数，对于选择用微信账户登录的用户，可以获取并展示其头像和昵称；“成就”中展示用户获得的成就，并提供分享功能，供用户将成就完成百分比分享至微信好友。



## 5.4 实现方案

### 5.4.1 前端技术

1. 使用npm对第三方组件库进行包管理。
2. 使用移动端组件库Vant Weapp。
3. 使用weapp.qrcode.esm.js来生成打卡所需要使用的二维码。

### 5.4.2 后端技术

使用微信云开发技术作为主要后端服务。

1. 使用云开发数据库作为数据库。
2. 使用云存储存储背景图片、成就图片等大图片。
3. 使用云函数进行数据库增删改查，返回数据上限更高，有不受限的读写权限。
4. 使用内容安全管理，对活动名和参与者昵称进行内容安全审核，减少审核及运营过程中的违规问题。

### 5.4.3 部分核心技术分解

#### 5.4.3.1 活动状态

难点：每一个活动会被所有的活动参与者访问，且活动的参与者有两种不同的身份权限（参赛者和管理员），当活动处于不同状态下，不同身份权限的人所看到的界面、可以发出的行动应该是不同的。

解决方法：为每一个活动设置了一个属性：state，每次用户进入活动详情界面时从数据库读取当前state。



state=2：管理员可以随时查看和改动打卡设置；参赛者看不到比赛路线。

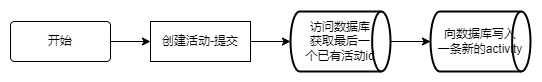
state=1：管理员不可改动打卡设置，仅可以查看；参赛者可以看到路线并打卡。

state=3：管理员不可改动打卡设置，仅可以查看；参赛者详情界面仅可以查看活动成绩，不可以进行打卡。

#### 5.4.3.2 ID自增

难点：管理员创建活动时对活动的名字不做限制，可以重复，那么为了访问某个特定的活动就需要编号。云开发数据库对每一条写入的数据只有固定的主码：“\_id”，且没有自增功能。

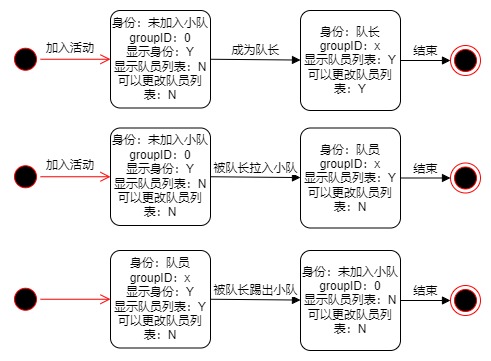
解决方法：为“activities”collection添加索引“id”，并在创建活动时进行如下操作：



#### 5.4.3.3 团队赛实现

难点：在完成个人赛后，迭代式实现团队赛功能，要在保证个人赛原逻辑不被修改的情况下增加团队赛不同身份的区分。

解决方法：根据赛制通过wx:if选择是否渲染团队列表view；根据团队赛中身份的不同渲染不同的团队列表view。



其中，队长可以拉入小队的备选者条件为活动ID为当前活动ID，身份为参赛者，groupID为0。有且仅有身份为队长的人可以编辑队员。

关于groupID，在创建活动时在数据库“groups”collection中添加一条id=当前活动编号，groupID=0的数据。当队长创建新的小队时进行类似5.4.3.2的逻辑的实现。

#### 5.4.3.4 列表实现

难点：列表缺少可直接使用的组件，需包含序号、用户名、路线、时间，且元素的值在数据库不同的表中。另外，当参赛人数较多时，界面的大小成为一大限制。

解决办法：通过view自建表格，利用微信开发者工具中的wx:for属性实现对列表的快速渲染。同时，利用scroll-view，实现了除表头外纵向滑动，即使数据很多也可以和表头一一对应，使用者体验良好。

#### 5.4.3.5 协调不同屏幕比例

难点：不同型号的手机屏幕比例不同，有些滚动视图的大小位置无法通过固定值设置。

解决方法：通过微信提供的接口getSystemInfoSync()和createSelectorQuery()来获取当前机型的屏幕高度和组件位置，并进行微调。

#### 5.4.3.6 打卡

难点：模拟传统电子打卡机器。

解决方法：小程序继承电子打卡器的CLEAR、打卡时间记录等功能，并增加了可视化的进度条。

（以扫码打卡为例）如果扫描的二维码不属于任何打卡点，则显示打卡失败。如果扫描的二维码属于某一打卡点，则显示打卡成功并提示当前的打卡点。如果打卡成功，但是不是预期的打卡点（即打卡顺序错误），则步骤条不推进。如果打卡点是正确顺序的打卡点，步骤条推进，并显示当前时间。

如果打卡记录为空，则成绩无效。如果打卡记录不是第一个为起点、最后一个为终点，则成绩无效。如果打卡记录中第一个为起点、最后一个为终点，但是没有正确的打卡序列，则成绩无效。如果打卡记录中第一个为起点、最后一个为终点，并且有正确的打卡序列，则成绩有效。

## 5.5 测试

### 5.5.1 功能测试

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试用例 | 种类 | 前置条件 | 期望结果 | 测试结果 |
| 创建活动 | 功能测试 | 无 | 数据库“activities”增加一条活动数据，活动id自增 数据库“actors”增加此管理员的数据 数据库“users”进行相应的修改 完成创建后，页面跳转到首页并显示该活动 | 通过 |
| 打卡设置 | 功能测试 | 活动已创建 | 数据库“information\_formanager\_backup”备份打卡设置，确保在任何一次提交后，都可以重新进入页面，在之前编辑的内容的基础之上再次编辑打卡信息 数据库“details”增加一条活动数据 数据库“pointsTable”增加一条活动数据 完成设置并提交后，页面跳转到首页 | 通过 |
| 编辑管理员 | 功能测试 | 活动已创建，并且有人加入活动 | 管理员人数有下限 数据库“actors”修改参与者的数据 | 通过 |
| 加入活动 | 功能测试 | 活动已创建 | 数据库“actors”添加参与者的数据 成功加入后，页面跳转到首页 参与者可以在首页的“未开始”中看到加入的活动 管理者可以在人员列表中看到活动参与者 | 通过 |
| 开始活动 | 功能测试 | 活动已创建 | 数据库“activities”修改活动的状态 打卡设置页面变为只读 参与者和管理者可以在首页的“正在进行”中看到该活动 | 通过 |
| 打卡 | 功能测试 | 活动已开始 | 参与者打卡后会有成功或者失败的提示 参与者可以通过扫描二维码或者输入密码的方式，进行打卡 数据库“results”添加一条打卡信息 数据库“testpage\_backup”备份参与者打卡数据，确保参与者退出打卡页面后，重新进入打卡页面打卡信息自动恢复 打印成绩单后，数据库“scorepapersTable”更新参与者的成绩 | 通过 |
| 成为队长 | 功能测试 | 活动赛制为团队赛，参与者已加入活动 | 数据库“groups”添加一个团队，团队id自增 数据库“actors”修改参与者的身份和groupid信息 | 通过 |
| 编辑队员 | 功能测试 | 参与者已成为队长 | 队员人数有上限 数据库“actors”修改队员的group信息 | 通过 |
| 结束活动 | 功能测试 | 活动已开始 | 数据库“activities”修改活动的状态 管理者可以在管理中看到所有参赛人员的成绩 参与者可以在活动详情页面看到自己的成绩 参与者和管理者可以在首页的“已结束”中看到该活动 | 通过 |

对系统中主要功能模块分别进行了单元测试，从结果可以认为小程序功能达到预期目标。

### 5.5.2 性能测试

于2022年5月28日进行了集成化性能测试，结果详见6.3，符合用户体验需要。

# 6. 应用运营

## 6.1 上线发布

目前已有上线版本，并更新至5.0.0。

## 6.2 运营维护方案

### 6.2.1 引流策略

1. 目前我们团队已经与南京大学定向越野队达成合作，“定向越野ING”这款微信小程序已经投入到了校定向越野队的日常练习与比赛中。作为校内唯一的马拉松长跑校队，校队的成员具有丰富的定向越野专业知识和技能，并且定期在校内开展定向越野训练，为小程序提供了天然的优质用户。
2. 同时，校内各院系也会为新生举办趣味定向越野，向刚迈入大学的新生进行推广，可以不断扩充用户人群。
3. 此外，还可以进一步推广，将“定向越野ING”推广给外校的定向越野社团，让小程序走出南京大学的校园，服务更多的人群。

### 6.2.2 存留策略

在获得用户后，针对增强用户的粘性，最大限度地满足用户的使用需求，提高用户的满意度，构建如下存留策略：

1. 优化打卡方式：增加GPS定位打卡功能。在参与者打卡时，通过GPS定位进行辅助判定，确保参与者是在目标地点打卡，防止活动中出现违规作弊行为，增强活动的公正性。
2. 增加活动的趣味性：增加趣味定向模式。在每个打卡点添加小游戏环节，通过团队合作完成小游戏，从而提升参与感。
3. 突出活动的积极价值：增加知识竞赛模式。在每个打卡点添加知识竞赛答题环节，根据活动的主题，从题库中抽取题目进行抢答。

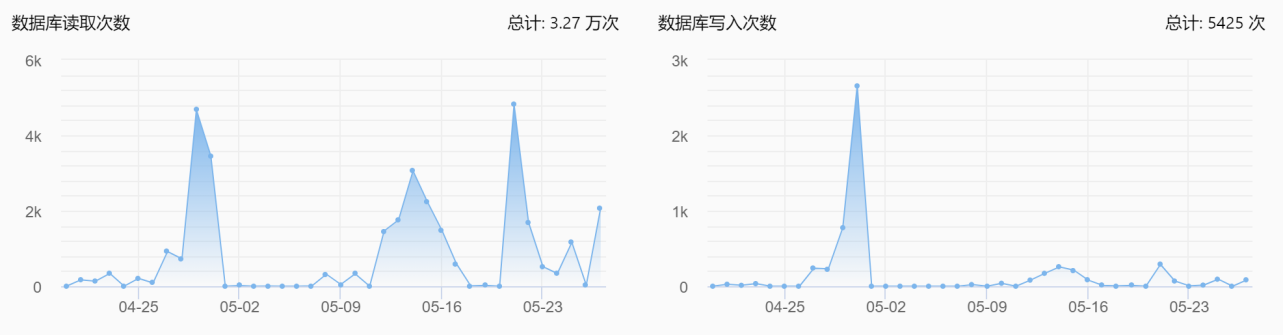
### 6.2.3 优化策略

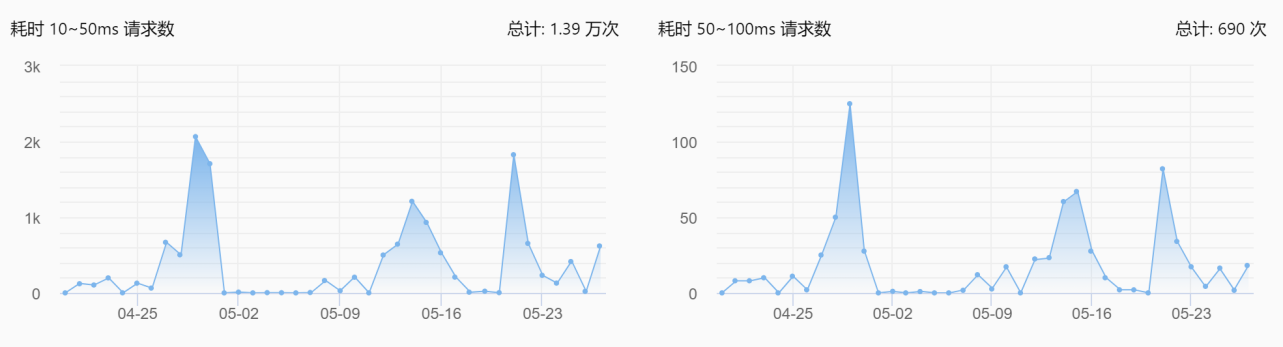
在一段时间的运营后，对相关数据进行分析总结，发现运营策略有如下可调整优化的方面：

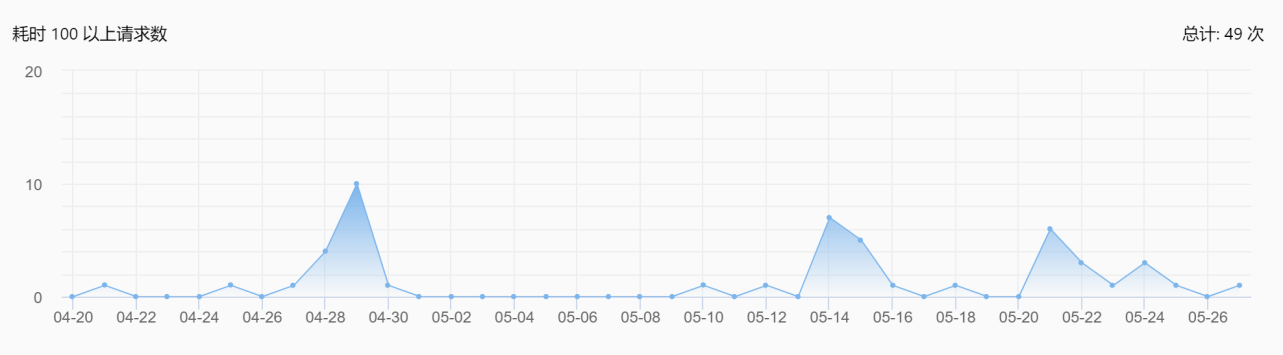
1. 优化正反馈机制。添加更多的个人成就和等级，参加的活动越多、成绩排名高，活动的成绩就越多，等级也越高。增加一个公开的排行榜，对于成绩优异或者积极参与活动的个人进行表彰，从而为参与者提供良性的反馈，让用户爱上运动。
2. 寻找赞助渠道。随着用户不断增加，活动规模不断提高，从而吸引赞助商赞助，为比赛提供定制的荧光手环等活动的奖品。

## 6.3 实际运行数据

以下图片来自云开发运营分析：







以下图片来自小程序性能分析：



# 7. 总结

在此小程序开发过程中，项目组成员基本是从零开始学习前后端技术，但是学习热情高涨，互相帮助，定期开会交流进度和技术，团结高效地完成了这个项目。在开发过程中，除了编程技术外，组员们还体会了微信生态开发技术的便捷，了解了一个软件的整体开发流程，学会了在项目中团队合作开发、代码管理的方法。项目组成员都表示受益匪浅。

大部分组员都是第一次接触定向越野活动，在开发过程中也领略了定向越野这一运动给参与者带来的健康与快乐，深刻感受到了运动健康之于自身、社会的重要性。在测试和交付合作方的过程中，受到定向越野队队员的肯定，组员也感到莫大的自豪和满足。这一切都激励了小组成员们不但要在未来继续维护并优化本程序，还要继续用自己的专业能力服务社会、服务人民，用科技创造社会价值，用智慧带来更美好的生活。