技术解决方案

### 方案概述

本方案主要服务于设计定向越野APP，应用中大部分功能如应用设计，应用的实现，服务器的搭建，服务器的访问将由我们完成，而部分功能如运营商的短信验证发送我们将使用mob公司提供的API,在数据回传方面我们使用个推的方案，降低服务器的访问量。

### 需求分析

本系统主要为定向越野比赛的组织者及用户提供软件支持，比如比赛组织及数据统计。不同于传统的运动软件主要服务于个人，本系统旨在给用户更多的选择，用户可以选择加入到官方举办的定向越野比赛，也可以作为定向越野比赛的组织者，方便的为其他用户提供服务。相对于传统定向越野比赛，减少了繁杂的工具，仅需要一台手机，既可以参加进来。

本系统主要服务于年轻的学生团体及部分对定向越野有所了解的社会人士。用户不需要有很高的学历，只需要有使用地图软件的经验，就可以很轻松的使用本系统，在地图软件如此普及的今天，使用我们设计的定向越野软件的学习成本将会十分的低。使用频度按用户日常习惯而定，按定向越野比赛的组织频率来说，大概会在每周一次左右。

### 系统总体设计

1. **界面设计**

整个软件采用底部导航栏，控制中间viewpager的内容显示，而viewpager的显示是由fragment填充的，所以fragment方面数据的填充非常重要，软件中出现了数据加载太多导致内存溢出的现象，针对这种情况，我们团队决定在以后采用网络缓存异步加载的方式来改善，并且，软件页面的适配工作是重点要改进的地方，尽量的多用percent布局，以力争能够在多种机型中能尽可能的达到美观的效果。

1. **登陆注册模块**

调用了mob公司的SDK，通过获取mob公司提供的免费验证码，获取验证码成功之后返回验证，再通过验证结果进行下一步操作，在这里的注册和更改密码都是用的同一个activity跳转，实现功能的不一样，是在登陆模块传入按钮的信息，从而发送不同的网络请求来进行修改密码和注册的行为。

1. **比赛的创建和加入**
2. 基于网络请求的方式，通过dialog中信息的填写，并且获得登陆用户的id，一起传输到服务器并新建比赛，比赛能否新建成功会通过网络请求的返回值进行判断，逻辑设计分布在服务器端以及安卓端。
3. 通过网络请求，获取已有比赛的队伍信息并加载在spinnner控件上，从而以供用户选择，若用户选择新建队伍，则软件会获取输入的名称以及队伍信息并一起上传到服务器，同时服务器端也会返回创建队伍的结果。
4. **服务器的搭建**

1). 框架选择:Django+uwsgi+nginx

2). Django:自带WSGI协议，用于处理Web服务器和应用程序的交互，但是性能一般，只能用于测试，它强调代码复用，多个组件可以很方便的以“插件”形式服务于整个框架通过django的特性，可以简便快速的开发出数据库驱动的网站，同时提供admin可视化数据查看界面，方便数据的管理。

3). Uwsgi：中间件

a) uWSGI本身是内网接口，开启多个work和processes可能也不够用，而nginx可以代理多台uWSGI完成uWSGI的负载均衡；

b) 支持的并发量更高;方便管理多进程；发挥多核的优势；

c) 提升性能，因为uwsgi协议比WSGI协议有优势

4). Nginx：HTTP和方向代理服务器：

a) 负载均衡

b) 安全：客户端对Web服务器的访问需要先经过反向代理服务器。这样可以防止外部程序对Web服务器的直接攻击。

c) 提升Web服务器IO性能，在反向代理服务器完整的接受请求，然后传给web服务器，保证服务器性能；

d) 单独处理静态文件的访问，这并不是uwsgi，django的强项

1. **服务器和程序的数据传输**

服务器和APP在通信时，由于讲的“语言”不通，服务器的数据需要可以被APP看懂，我们使用了JSON格式传输数据，并在刚开始定下输入输出，在APP制作中间增删查改，保证APP收到的数据准确无误。

### 设备清单

1. 租用的服务器一台：VULTR提供
2. 四台测试调试用ANDROID机器
3. 四台编程用计算机