# 产品实现方案

目录

[1系统的主要功能 1](#_Toc18502)

[2UI界面设计 2](#_Toc28782)

[3关键技术和技术难点 2](#_Toc31272)

[3.1自动识别运动状态(可在后台Service保活) 2](#_Toc3566)

[（1）健走的数据 2](#_Toc27440)

[（2）跑步的数据 3](#_Toc18270)

[（3）骑行的数据 3](#_Toc24036)

[3.2绘制运动轨迹 3](#_Toc17489)

[3.3利用Bmob建立云端数据库 4](#_Toc20867)

[利用Bmob建立云端数据库 4](#_Toc27952)

[难点：Bmob的使用人数不多，开发文档不够详细 4](#_Toc589)

[3.4绘制历史数据图表 4](#_Toc13496)

[4用户体验记录和分析 4](#_Toc2467)

[（1） 界面美观性和界面设计合理性用户体验效果良好 4](#_Toc16913)

[（2） 跑步和健走的识别准确率很高，但是骑行的识别准确率较低 4](#_Toc16426)

[（3）程序运行速度和效率良好，实现了系统等方面的优化 4](#_Toc10466)

[（4）程序设计无亢杂，功能简介明了 4](#_Toc31267)

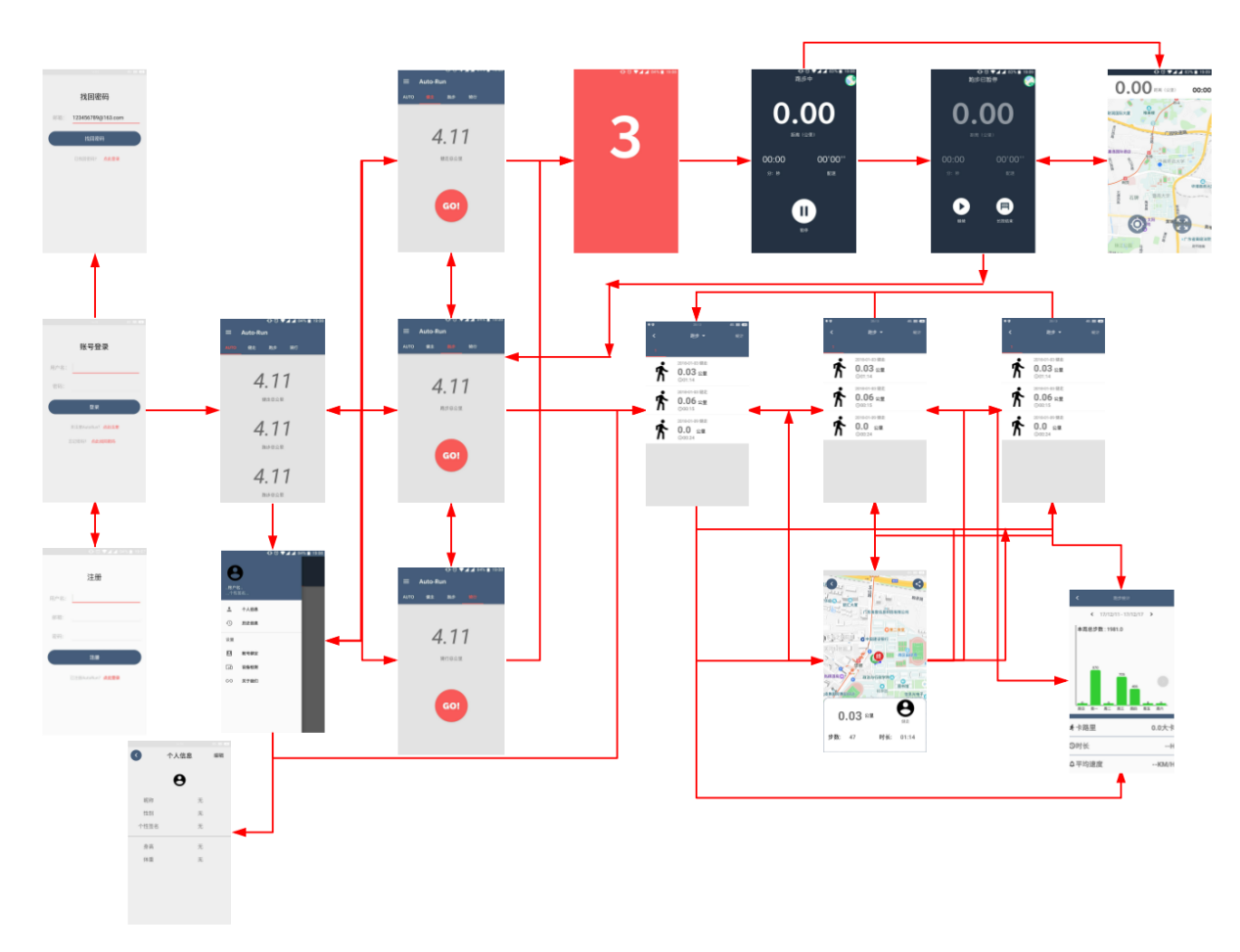
[（5）没有实现用户间互动的功能，无法分享用户的运动心得 4](#_Toc1750)

[5已完成的改进和存在的问题 5](#_Toc14157)

# 1系统的主要功能

1. 通过邮箱注册登录。
2. 可编辑用户个人信息。
3. 把app放到后台可实现自动识别运动状态（健走、跑步、骑行）并记录健走和跑步的步数以及骑行的里程数，当程序不在后台也不在前台就不会记录，实现了每天零点更新。
4. 可记录一周的健走、跑步、骑行的数据，通过表格显示，每周六零点更新。
5. 用户可以自由选择详细记录一次健走、跑步、骑行的数据（健走和跑步可提供运动轨迹、步数、里程数、持续时间，骑行可提供运动轨迹、里程数、持续时间），这里记录的数据会上传到云端数据库并保存下来。
6. 查看历史数据记录，可查看以往用户记录的详细运动数据，其中运动轨迹可在可以滑动的地图上显示。

# 2UI界面设计



# 3关键技术和技术难点

## 3.1自动识别运动状态(可在后台Service保活)

计步点的功能是学习了相关开源代码之后写出来的，而网上的关于识别运动状态的论文通过实践发现并不适用，于是只好尝试着自己记录加速度的数据，通过发现健走、跑步、骑行相关数据的差别，写出了一个识别准确率较高的算法。

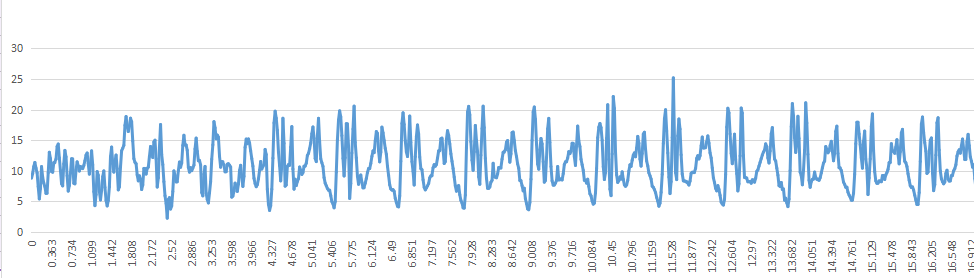
下面是xyz轴的加速度的平方和随着时间的变化的表格

（1）健走的数据

方差：3.7

波峰取值：16—25

波峰波谷时间间隙：0.25秒

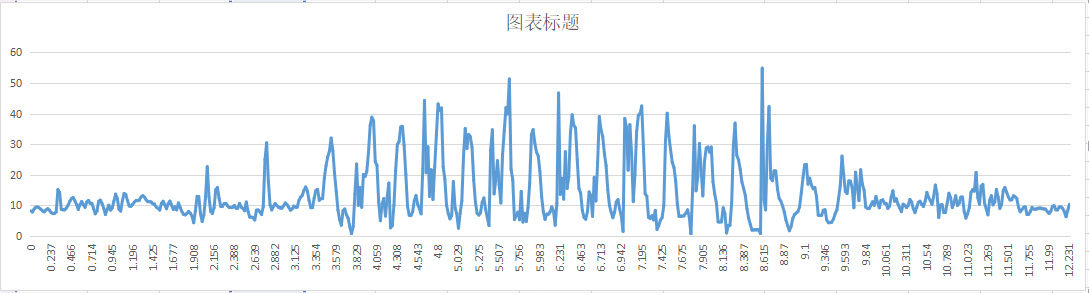


（2）跑步的数据

方差：11.3

波峰取值：30—55

波峰波谷时间间隙：0.25秒



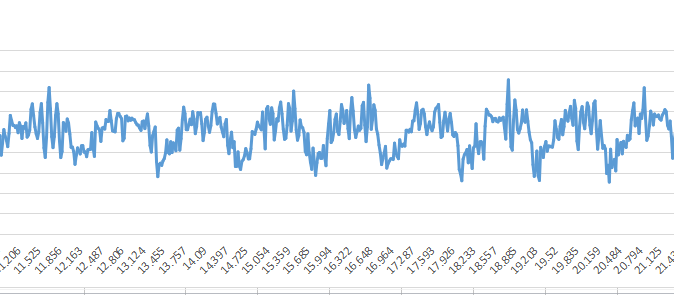
（3）骑行的数据

方差：1.9

波峰取值：13—16

波峰波谷时间间隙：0.45秒

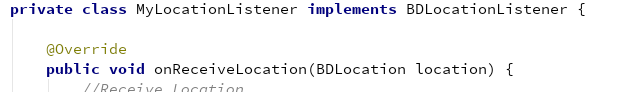
（识别骑行的方法通过与百度地图定位相结合计算出速度以提高准确率）



通过分析表格，可以发现这三种运动的数据的方差、波峰取值范围和波峰波谷的时间间隙有很大的差别，我们就是用这三个特征值区分出这三种运动状态的。

## 3.2绘制运动轨迹

通过调用百度地图api实现，从BDLocationListener的onReceivelocation获取当前位置的经纬度，频率是1秒1个，选取距离大于5米小于400米的位置点记录下来然后通过将记录下来的位置点绘制轨迹线条图层，显示在地图上。



由于百度地图api的学习成本很高，在看懂怎么用这个api上花了不少时间。

## 3.3利用Bmob建立云端数据库



**利用Bmob建立云端数据库**

云端数据库总共有3个表，一个用户个人数据表，一个简略历史运动信息表以及一个详细历史运动信息表

**难点：Bmob的使用人数不多，开发文档不够详细**

Bmob上手很容易，但是由于使用者不是特别多，然后官方文档有些问题又没有说明，导致如果出了意料之外的问题，只能自己尝试解决。比如Bmob自动导入的SDK和百度地图的SDK有冲突，而且官方没有提供解决方案。还有，它的查询提供的是异步的回调方法，但是官方开发文档并没有说明这一点，等等。

## 3.4绘制历史数据图表

根据时间戳和关键字获取本地preference的历史一周的健走/跑步/骑行数据。

基于自定义view计算并绘制柱形统计图，没用网上的开源框架，所以难度比较大。

# 4用户体验记录和分析

目前用户体验量不是很大，得出的体验结果不是很全面，目前主要有以下几点：

1. 界面美观性和界面设计合理性用户体验效果良好
2. 跑步和健走的识别准确率很高，但是骑行的识别准确率较低

（3）程序运行速度和效率良好，实现了系统等方面的优化

（4）程序设计无亢杂，功能简介明了

（5）没有实现用户间互动的功能，无法分享用户的运动心得

# 5已完成的改进和存在的问题

已完成的改进：

1. 自动识别骑行一定要启动Gps，然后结合手机移动的速度来判断。
2. 增加找回密码功能。
3. 查看详细历史信息的运动轨迹原本是一张图片，现在改进成在动态地图上显示。
4. 修复了数据小数位数显示出错、数据单位不符的问题。
5. 修复了表格数据下面的卡路里数据始终是零的问题。

仍然存在的问题：

1. 最大的问题是没有利用软件工程的思想将百度地图定位的相关函数封装利用起来（其实对于怎么封装还没有头绪，因为代码耦合度较高）。
2. 对于骑行的运动识别准确率较低，当跑步的时候有可能会识别到骑行。因为用户对于自己有没有骑行非常清楚，而出现这种识别错误非常影响用户的使用体验。
3. 当程序在后台运行的时候，用户重新点击桌面的图标启动的话会销毁后台运行的进程然后重新启动一个进程。（应该是可以完善的，但是由于时间有限所以没有解决这个问题）

（4）UI设计方面，我用的listview然后重写它的Adapter的方法实现复合信息的显示，但是其实现在很多应用都用recycleView来替代Listview了，它可以实现很多更好看的效果。

（5）功能方面，一开始写的设计文档里面，我们是有写删除历史记录功能和通过spinner查看不同类型的运动的记录，这两个功能的。但是由于使用了Fragment，然后时间也不太够，所以这个两个功能没有加上，这是功能上的不足之处吧。

（6）用表格显示数据的界面没有适配手机屏幕导致部分机型在这个界面无法下滑。