Fitness Application for Everyday Use

Omkar Salvi
School of Computing, Informatics and Decision Systems Engineering
Arizona State University
Tempe, Arizona 85281
Email: osalvi@asu.edu

Saurabh Jagdhane
School of Computing, Informatics and Decision Systems Engineering
Arizona State University
Tempe, Arizona 85281
Email: sjagdhan@asu.edu

Vishal Srivastava
School of Computing, Informatics and Decision Systems Engineering
Arizona State University
Tempe, Arizona 85281
Email: vsriva10@asu.edu

***Abstract*—In todays world, health and fitness is an important thing. Everyone wants to have a healthy and fit body. So to help user be fit and motivate him for daily exercise we have developed a fitness application. Our application helps the user to track his calories burnt, his exercise, his day to day work. Also it suggests the user what exercise he should do depending on the desired calories user want to burn.**

***Keywords*—Fitness App, Calorie, Context aware.**

I. Introduction

The application aims at tracking the users activity and reading the data from that movement and calculating the different parameters such as calories burnt, distance covered etc. by the user based on the data read. We are using sensors like Accelerometer, GPS, Proximity sensor to track the users activity and then doing appropriate calculations based upon the data read and the data entered by the user such as name, age, weight, height and sex to calculate the calories burnt by the user. The application will provide you with your daily calorie burnt, best exercise stats, will track your push-ups and run individually and will store the result in database for future use.

II. Implementation Details

A. Prediction of Exercise

For this we depended on accelerometer to judge the kind of exercises a person did. Here we calculated the resultant vector of accelerometer along x, y and z axis and passed through a first order low pass filter. After filtering we sample the resultant at each 500 msec and based on certain threshold decide whether a person is walking or running. The model is based on the difference in vibrations/accelerations in different exercises. When we are walking our motion of body is regular and we experience less amount of acceleration but when we are running the vibration/acceleration of body is of much higher magnitude. So from this difference in acceleration we can judge if a person is walking or running. It is more energy efficient and instantaneous as we are using accelerometer rather than GPS for prediction.

B. Application wide control

In order to enforce a system wide control mainly for activity 'Settings' we created a global Application class for our application. Certain controls or modes that are enforced by Settings are difficult to pass through Intents, specially when the concerned activity is three or four activities below in hierarchy. We can pass or stream Intent to the activity but its difficult to ensure no corruption of Intent when it makes pass through other in-between activities. So in order to enforce direct control to application we made a Application class to provide with global state variables that are only accessible through public get and set functions. Through this we didn't need to maintain the flow of data in the hierarchy and also provides us with a simpler model. It not only reduced the size of Intents passed and gave faster accessibility to state variables but also made it safe from any kind of in-between corruption of data.

C. Location Tracking for Running, walking and cycling

The locations obtained by the network are not very accurate and have some discrepancies. So we decided to use GPS to track the mobility for better results. On clicking the start button, location data is requested from GPS. If GPS is not activated in the smart-phone then, application takes the user to GPS settings, so that GPS can be activated. The application provides a timer as well which will start on clicking the start button. The timer will keep on counting until the user clicks stops button. The GPS will be updating the position of the user every 2 seconds. Also after getting the first location, application gives a toast called First Location. After every 2 seconds application will calculate the distance covered in recent 2 seconds and add it to the total distance covered. Following function is used to calculate the distance covered:

$$Location.distanceBetween(lattitude1, longitude1, lattitude2, longitude2, result)$$

It will calculate the distance between 2 positions given by (latitude1, longitude1) and (latitude2, longitude2). After the user has stop running, the total distance covered will be already calculated. Using total distance covered and total time spent average speed is calculated.

D. Calorie calculation model

We are using the Harris Benedict Method to calculate the calories burnt by the user in different physical activities. The HarrisBenedict equation (also called the Harris-Benedict principle) is a method used to estimate an individual's basal metabolic rate (BMR). The estimated BMR value is multiplied by a number that corresponds to the individual's activity level. This can be used to calculate the estimate of the calories burnt by an individual as per the physical activity he performed. Following are the equations used to calculate estimation of calories burnt by the individual

$BMR\ men = 66 + (6.23 \times weight\ in\ pounds) + (12.7 \times height\ in\ inches) + (6.8 \times age)$
$BMR\ women = 65.5 + (4.35 \times weight\ in\ pounds) + (4.7 \times height\ in\ inches) + (4.7 \times age)$
$PAL = Activity\ coefficient \times Activity\ time$
$Calories\ burnt = BMR \times PAL$

E. Proximity sensor for push-ups

We have used proximity sensor to count the push-ups performed by the user. The value of the sensor changes depending on the distance between smart-phone and user. The threshold was set as the minimum distance between smart-phone and user. Whenever the sensor senses that the distance is less than this min distance , it will increment the push-up count. So user has to put the smart-phone on ground facing upwards and perform the push-ups. When user clicks start button the app starts counting the push-ups. And when he clicks stop button the app stops counting push-ups.

F. Music Application Integration

It has become a common habit among users to listen to music during a workout or exercise and so in order to save user for managing two applications side by side that is Music Player and our Fitness App we have provided a integrated solution for this. We have incorporated a Music player that starts automatically when a user starts his exercises and stops when user is done with his exercise. Currently the app is made to play a single music file and when user is done with his activity and presses Done then music stops. This is provided as an optional feature which can be turned on/off using Settings. So, by this we save user from hassle of managing a different music app before and after an exercise.

G. Context aware suggestion

The unique thing about the application is it reminds you to start exercise activity and recommend about indoor or outdoor exercise depending on weather conditions once the user clicks on notification. For that GPS data is used to obtain latitude and longitude which are then passed onto geocoder to obtain current location of the user. The location is used for obtaining current weather conditions such as temperature, and weather conditions. Location of a user is obtained and then data is passed onto a Yahoo Weather API which indeed give JSON response to the query.

```
Geocoder gcd = new Geocoder(this, Locale.getDefault());
addresses = gcd.getFromLocation(lat,lng,1);
return addresses.get(0).getLocality();
```

The suggestions can be customized even further if there are any emergencies like dust or snow storm, windy, etc. conditions

H. Database

We are using SQLite database to keep track of user activities. We are creating the tables in database using the name, age and sex of the user. After creating the table application will keep on storing the daily activities into the table. We are storing the user activity statistics into the database table for recording, later retrieval and for user view. The database table will store the time-stamp, distance covered, average speed, calories burnt, target calorie burn and number of push-ups by the user on a particular day. The database will store the entire history of the user. The statistics for the last seven days are shown to the user in form of bar graph. This bar graph will show both desired and actual calories burnt by user so user can keep track of his performance for past few days. Also the best stats are shown to the user on the Scoreboard screen of the application.

I. Workout Planner

Deciding which exercise to do on a particular day is a crucial question for everyone. To solve this problem our application suggests the activities user should do to reach the target calories burnt. It will use the Harris Benedict Method to predict the calories burnt by the user in different physical activities according to their weight, height, age, sex. Using this method application will calculate approximate time for which user should perform different activities and it will suggest the user to perform the activity for that much time period. This will motivate user to exercise for at least suggested time period. Thus users confusion about which activity to perform each day will be solved by this application.

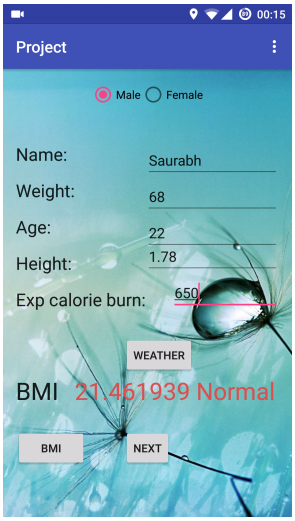
III. Task Completed

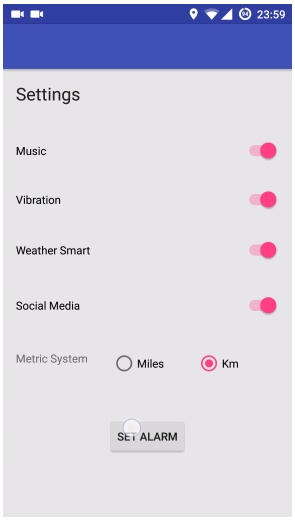
All the below tasks have been completed and are provided in the application.

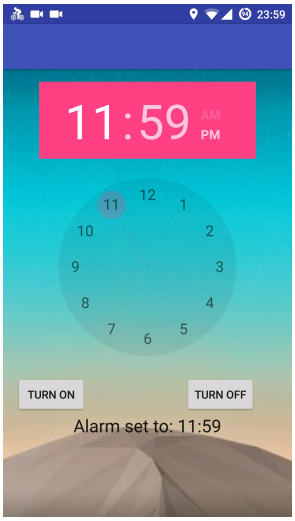
TABLE I. Work Division

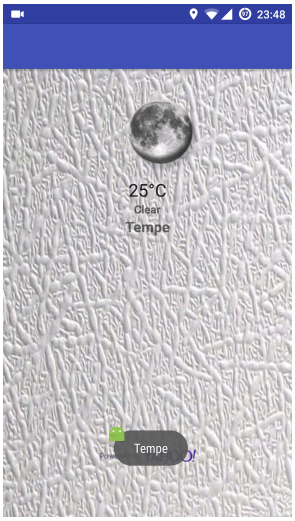
S.no	Task	Assignee
1	Weather API	Saurabh
2	Alarm notifications	Saurabh
3	Accelerometer and low pass filtering	Vishal
4	Prediction of exercise mode	Saurabh
5	Distance covered using GPS	Omkar
6	Proximity sensor	Omkar
7	Display users activity in a graph	Vishal
8	Scoreboard	Omkar
9	Database	Omkar
10	Notification through vibration	Vishal
11	Share on social media	Saurabh
12	Nutrient chart	Saurabh
13	Music application	Vishal
14	Settings	Vishal
15	Workout planner	Omkar

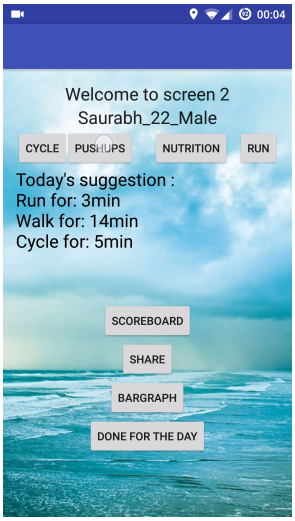
IV. Screenshots

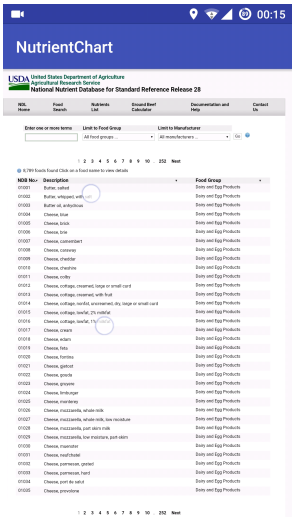


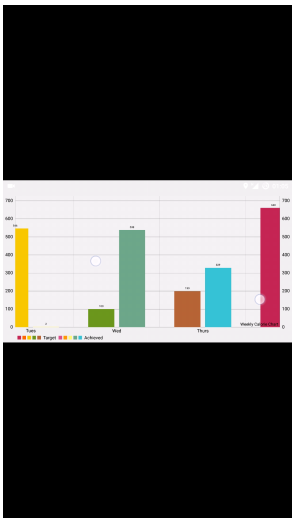












V. Conclusion

The main purpose of the application is to remind user to carry out exercise activity on a daily basis and suggest whether to work out in a gym or outdoor depending on a context. The calorie burnt by user through various exercise modes were saved in database successfully only if user wishes to and not upon exit through activity. As sensor data is necessary to be calibrated before user starts an activity. In many cases sensor data proved to be unreliable. To distinguish between various exercise modes it is necessary that device is placed on a user's arm regularly to gather proper sensor data. It has been observed that the application is unable to acquire GPS data on a very first attempt. To get notification for the exercise on the next day it is necessary that user closes application properly from task manager.

Acknowledgment

We would like to thank our Professor Dr. Ayan Banerjee for giving us an opportunity and guidance to create an android application which can be a part of daily life. We are also thankful to our TA Junghyo Lee for the invaluable knowledge, feedback and suggestions given during every phase of a project.

References

[1] https://thenewboston.com/videos.php?cat=6

[2] http://android-developers.blogspot.com/

[3] http://developer.android.com/guide/topics/location/strategies.html

[4] https://developer.yahoo.com/weather/

[5] http://code.tutsplus.com/tutorials/add-charts-to-your-android-app-using-mpandroidchart–cms-23335

[6] http://www.bmi-calculator.net/bmr-calculator/harris-benedict-equation/

[7] http://developer.android.com/training/basics/data-storage/databases.html

[8] https://ndb.nal.usda.gov/ndb/