Group 1: Fitness Evaluator Application Requirements Document

Fernando Barrios (1320297)
Cole Blanchard (1226905)
Ratna Emani (1310641)
Jay Nguyen (1327828)
Jade Orkin-Fenster (1335823)

March 6, 2016

McMaster University 3A04

Contents

1 Introduction

This project will focus on the design and implementation of a health application for a mobile device, in particular a device based on an Android platform. The main task of the application will be to evaluate the user's fitness from the perspective of three main experts, a dietician, a personal trainer and a physician. The user will be evaluated by each expert separately and the results will be combined to perform an overall evaluation of the user's lifestyle.

1.1 Purpose

The purpose of this Software Requirements Specifications *(SRS)* document is to provide a detailed description of the requirements for the software to be. This *SRS* will allow a complete understanding of what is to be expected from the application. A clear understanding of the application's functionality is vital to help develop correct software for the end users and to develop future stages of the project.

This document will be used by a variety of engineers, consultants and stakeholders. This document will allow engineers to fully understand the scope of the project and develop a solution accordingly. The consultants will also benefit from the document to increase their understanding and ensure that business needs will be met. While stakeholders will be able to review the document to ensure that the system will be built to their expectations and make changes accordingly.

1.2 Product Scope

The scope of this project will be to specify, design and implement a fully functional application for a mobile device with an Android platform. The application will provide the user with an evaluation regarding their health based on their diet, fitness level and their body mass index. The values will be an estimate, but together shall provide an adequate evaluation of the user's overall health. The goal of this application is to allow those who are interested in improving their overall fitness level to track their daily progress. The application will also encourage users to improve their fitness levels as they learn how to improve their fitness rating. The application will be a stand alone application, however the fitness expert will interface will Google Maps to evaluate the user's cardiac capacity.

1.3 Definitions

This section shall contain definitions used in this document with respect to the Fitness Evaluator. The definitions displayed in this section are specific to this document and may not be identical to definitions used elsewhere. The purpose of this section is to help the user understand the references in this document and to fully understand the requirements. The functional requirements are specifications of the products functionality (what the product must do), such as calculations, whereas the non-functional requirements describe system attributes, such as security, look and feel.

Term	Definition
BMI	Body Mass Index
Cooper Test	An estimation of VO2 Max based on distance run and time taken
GPS	Global Positioning System
GUI	Graphical User Interface
Rockport Fitness Walking Test	An estimation of VO2 Max based on distance walked, time taken, gender, age, weight, and heart rate
SRS	Software Requirements Specifications (Refers to this document)
VO2 Max	Maximum Volume of Oxygen
WC	Waist Circumference

1.4 References

Andrew LeClair.
SE 3A04: Requirements Templates
Department of Computing and Software, McMaster University, January 27/28, 2016

1.5 Overview

The *SRS* will be organised into three main section. The first section is "The Overall Description", the second section is "Functional Requirements" and the final section will be the "Non-Functional Requirements". The overall description will describe the requirements of the application, while the final two sections will list in detail the requirements of the system.

2 The Overall Description

This section shall describe the general factors which will effect the system. This section will provide a background for the specific requirements from section three, and makes it easier to understand them.

2.1 Product Perspective

There is an increasing awareness of personal fitness and healthy habits in today's technology-driven society. One of the biggest markets for fitness is now the App Stores for many mobile phones. There are APIs designed to track, measure, and time all physical activities the user engages in with their phone. Currently the App Store is populated with products similar to Nike+ or NexTrack, which track steps and provide the user with helpful guides and tutorials for various exercises. Recently, there has been a rise in fitness-oriented hardware, which the users can wear to track and analyse an even greater amount of data. Some of the more common products include FitBit, DigiFit, and Jawbone.

We would like to design an application that ideally fits in-between these two worlds. The application would have much of the functionality of the wearable hardware, without

requiring the actual equipment. We can achieve this by designing an application that would be able make more accurate predictions using a wider array of data. In contrast with common market applications that specialize in a single expertise, our application would be able to use up to 3 experts at a time to make calculated predictions about the user's progress and goals.

2.2 Product Functions

2.2.1 Physician

F1. The system will provide a *BMI* measurement based on user input of weight and height, and will refine this measurement if the user inputs gender and/or age.

F2. The system will provide an estimation of body fat percentage based on user input of height and waist size.

F3. The system will make a recommendation for whether the user should try to lose, gain, or maintain their current weight based on their *BMI* and body fat percentage.

Additional Features

F4. The system shall compare the user's *BMI* measurement to other people in the region based on *GPS* data and government demographics data and inform the user how they compare to the general population in their area.

2.2.2 Personal Trainer

F5. The system will store the user's information.

F6. The system will calculate calories burned from walking, running, or biking along a path.

F7. The system will assess cardiovascular fitness by calculating *VO2 Max* by the *Cooper Test* or the *Rockport Fitness Walking Test* methods, based on distance run or walked respectively, and the length of time this took.

Additional Features

F8. The system will allow the user to select a biking/running/walking path based on its difficulty in terms of elevation increases, using user inputs of the length of time they wish to spend on their exercise and/or the calories they wish to burn during their exercise. How long a path will take the user and how many calories they will burn will be based on their personalised previously-assessed cardiovascular fitness.

F9. The system will provide difficulty, length of time, and/or calories-burned information for marked bike and walking trails based on the user's fitness level. These numbers will originally be pre-set average data, but will be continuously updated based on length of time and distance travelled along the route.

F10. The system will track the user's progress by saving their cardiovascular fitness when they use the application and tracking their improvement over time.

2.2.3 Dietitian

F11. The system will recommend to the user how many calories they should be consuming per day to gain, lose, or maintain their current weight. This recommendation will depend on the Physician's recommendation.

F12. The system will check if the number of calories that the user consumed (based on user input) is within the range for gaining, losing, or maintaining their current weight.

F13. The system will determine the number of calories that the user consumed in a day from the user entering the food items they ate. A database will be used to look up how many calories the food item contains.

Additional Features

F14. The system will allow the user to enter food items by bar code scan and will retrieve the food's caloric content and nutritional information.

2.3 User Characteristics

There are two types of users that interact with this system: The general user, and the software developer.

2.3.1 The General User

The general user is expected to be familiar with the *GUI* elements of smart phones and is to be able to read and comprehend English. In addition, they are expected to know basic knowledge about their body, such as weight and height and understand the basics of food nutrition.

2.3.2 The Software Developer

In addition to the points above, the software developer is expected to have knowledge on object oriented programming, and rudimentary knowledge with working on Android Studio.

2.4 General Constraints

The general constraint of the project is time constraint as there is no budget required for the design and implementation. The allocated time for the project with be roughly two months with a total of four deliverables to ensure thorough completion of the application and its documentation.

2.5 Assumptions and Dependencies

The personal trainer expert will depend on the availability and accuracy of Google Maps.
The personal trainer expert will depend on access to the internet to access Google Maps.
The physician expert will depend on the availability and accuracy of Google Maps.
The physician expert will depend on access to the internet to access Google Maps.
The physician experts will depend on the accuracy of census data.
The hardware will need to be able to use the Android operating system.

2.6 Apportioning of Requirements

The system shall allow all profiles to be stored in a central database. Using the database, the level of fitness can be more accurate for each user.
The app can be run on different operating systems.
Developers can add their own experts which will be stored in a central database. Users search for experts and can then download/remove experts to fit their preferences.

3 Functional Requirements

BE1. The app is opened

VP1.1 User

i. The system shall allow the user to create a profile.
ii. The system shall give a selection of experts from which the user can choose from.
iii. The system shall allow the user to select "How Fit are You?"

VP1.2 Developer

i. The system shall allow the developer to add and remove experts from the application.

BE2. Edit profile is selected

VP2.1 User

i. The system shall allow the user to enter their age, weight, gender, height, and waist measurements.

BE3. Personal Trainer expert is selected

VP3.1 User

i. The system shall calculate the distance of the user's jogs using Google Maps.

ii. The system shall calculate the time and speed of the user's jog.

iii. The system shall calculate the user's *VO2 Max*.

BE4. Physician expert is selected

VP4.1 User

i. The system shall calculate the *BMI* of the user based on their profile.

ii. The system shall locate the user using Google Maps.

iii. The system shall give a relative *BMI* rating based on the region the user lives in using Canadian census data.

BE5. Dietician expert is selected

VP5.1 User

i. The system shall allow the user to select food from a database.

ii. The system shall allow the user to manually enter calories.

iii. The system shall calculate the total calories the user consumed daily.

VP5.2 Developer

i. The system shall allow the developer to update the food database.

BE6. "How Fit are You?" is selected

VP2.1 User

i. The system shall identify how fit the user is using at combination of at least one or more experts.

4 Non-Functional Requirements

4.1 Look and Feel Requirements

4.1.1 Appearance Requirements

LF-A1. The application shall display a *GUI* with buttons and dialogue boxes.

LF-A2. The application shall display a notification if a required option that is needed to perform a requested process has not yet been set.

LF-A3. The application shall display which options have not yet been set before a process is requested.

4.1.2 Style Requirements

LF-S1. The application should use a simple colour scheme that is consistent across all pages and screens.

4.2 Usability and Humanity Requirements

4.2.1 Ease of Use Requirements

UH-EU1. The application shall be easy for users to install.

UH-EU2. The interface shall be intuitive and easy to use for users with a secondary-school education.

UH-EU3. The application shall make clear to the user what input information is necessary to perform a task and what input information is optional that will improve the accuracy of the fitness estimation.

4.2.2 Personalization and Internationalization Requirements

UH-PI1. The system shall store the user's data from one session to the next and may archive old data instead of overwriting it for the purpose of progression tracking.

UH-PI2. The system shall support user input and data output in both the metric and imperial systems.

4.2.3 Learning Requirements

UH-L1. The application shall not require a tutorial and shall communicate enough information through its *GUI* to guide users to successfully completing desired actions.

UH-L2. The system shall provide the user with the ability to read the definitions of specialised terms that are used in the application (e.g. *VO2 Max*).

4.2.4 Understandability and Politeness Requirements

UH-UP1. The application shall only use icons for its buttons that are clearly recognizable to user (including young users who are not familiar with outdated technology) and associated with the action they represent.

UH-UP2. The system shall use terminology understandable to users with a secondary-school education.

UH-UP3. All text displayed by the interface shall be in English with Canadian spelling.

4.2.5 Accessibility Requirements

UH-A1. The application should not use red in contrast with green to avoid issues for users with red-green colour blindness.

UH-A2. The application shall use larger, readable fonts.

4.3 Performance Requirements

4.3.1 Speed and Latency Requirements

PR-SL1. The application shall launch within 2 seconds.

PR-SL2. The system shall be responsive and when all required criteria to perform an action are met, it shall perform the requested action within 10 seconds.

PR-SL3. The system shall be responsive and when not all the required criteria to perform a requested action are met, it shall display a notification of the error within 2 seconds.

4.3.2 Safety-Critical Requirements

PR-SC1. The system shall not provide encouragement to diet or dieting tips to users who have been determined to be underweight.

PR-SC2. The system shall not provide users with dangerous walking/running/biking routes and instruct them to follow these routes (e.g. across a highway.)

4.3.3 Precision or Accuracy Requirements

PR-PA1. All calculations within the system based on user input and phone sensor input shall be precise to two decimal places.

PR-PA2. The system shall be adaptable to differing amounts of input data where appropriate, such that it is able to calculate a rough estimate of a fitness criterion with the basic information, but is able to provide a more accurate estimation of the the criterion the more input data it is provided.

4.3.4 Reliability and Availability Requirements

PR-RA1. Even if one expert fails, other experts shall be independent enough to not fail as a result as well.

PR-RA2. The application shall be reliable and only unexpectedly quit at a failure rate of 5

4.3.5 Robustness or Fault-Tolerance Requirements

PR-RF1. The system shall still continue to operate despite a loss in internet connection, even though not all features will be available.

4.3.6 Capacity Requirements

PR-C1. The system must be able to store and/or access a searchable database of nutritional information for at least 1000 of the most commonly use food items.

4.3.7 Scalability or Extensibility Requirements

4.3.8 Longevity Requirements

PR-L1. The product shall be expected to be operational for a minimum of one year.

4.4 Operational and Environmental Requirements

4.4.1 Expected Physical Environment

OE-E1. The application shall run on Android devices with *GPS* tracking, photograph sensors, accelerometer sensors, gyroscope sensors, and internet connection.

4.4.2 Requirements for Interfacing with Adjacent Systems

OE-I1. The system shall be able to read and parse *GPS* and sensor data from the Android phone with which it interfaces.

OE-I2. The system shall be operational on all Android phones running at least the three most recent operating system releases.

OE-I3. the system must interface with *APIs* running on remote servers.

4.4.3 Productization Requirements

OE-P1. The product shall be downloadable for Android phones.

4.4.4 Release Requirements

4.5 Maintainability and Support Requirements

4.5.1 Maintenance Requirements

MS-M1. All methods and modules within the system shall be documented with comments within the code to facilitate easy maintenance and readability.

MS-M2. The system shall contain mechanisms by which a system administrator may update databases contained within the system.

MS-M3. New experts shall be possible to add to the system within one day by a trained developer who is familiar with the system documentation.

4.5.2 Supportability Requirements

4.5.3 Adaptability Requirements

MS-A1. All experts shall be separable from one another and easily swappable.

MS-A2. The system shall be sufficiently modularised such that expert modules can be added or removed without requiring extensive system restructuring.

4.6 Security Requirements

4.6.1 Access Requirements

4.6.2 Integrity Requirements

SR-IN1. The system shall encrypt all transmitted messages using a cryptosystem using symmetric encryption with a Vigenere cipher.

4.6.3 Privacy Requirements

SR-PR1. The system shall not transmit, upload, or otherwise disclose the user's personal information without the user's permission.

4.6.4 Audit Requirements

4.6.5 Immunity Requirements

4.7 Cultural and Political Requirements

4.7.1 Cultural Requirements

CP-C1. The application shall not contain any imagery or text that can be reasonably foreseen as potentially offensive to users by mocking, insulting, or appropriating their culture, cultural symbols, cultural experiences, or cultural background.

CP-C2. The application shall not shame, insult, or guilt users for their body shape, body size, or fitness level. The system shall not use negative adjectives to describe the user's body, fitness, or health (e.g. "poor fitness", "bad health", "unhealthy weight"). Whenever a comparison is performed to demographic or standard data and the user is found to be below average, this should be contextualised with positive information reported as well (e.g. "Your *VO2 Max* is better than 25 percent of the population!", "You have achieved a 5 percent reduction in *BMI* measurement!")

4.7.2 Political Requirements

CP-P1. The application shall not contain any content that can be reasonably seen to favour or criticise any Canadian political party or politician and shall remain politically neutral.

4.8 Legal Requirements

4.8.1 Compliance Requirements

LR-C1. The system shall comply with all Canadian federal laws and all Ontario provincial laws.

4.8.2 Standards Requirements

A Division of Labour

Team Member	Contribution
Fernando Barrios	Introduction, Contribute to Requirements
Cole Blanchard	Functional Requirements, Assumptions and Dependencies, Apportioning of Requirements
Ratna Emani	Product Perspective
Jay Nguyen	User Characteristics
Jade Orkin-Fenster	Product Functions, Non-Functional Requirements

Group 1 Signatures

Fernando Barrios

Cole Blanchard

Ratna Emani

Jay Nguyen

Jade Orkin-Fenster