

Software Requirements Specification

for

Buff Stuff

Version <1.0>

Prepared by

Group Name: Swol Patrol

|  |  |  |
| --- | --- | --- |
| <Henry Unruh> | <11686704> | <Henry.unruh@wsu.edu> |
| <Thomas Bailey> | <11680885> | <Thomas.k.bailey@wsu.edu> |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |
| --- | --- |
|  |  |
| Date: | October 25th 2019 |
|  |  |
|  |  |
|  |  |

Contents

Revisions iii

1 Introduction 1

1.1 Document Purpose 1

1.2 Product Scope 1

1.3 Intended Audience and Document Overview 1

1.4 Definitions, Acronyms and Abbreviations 1

1.5 Document Conventions 1

1.6 References and Acknowledgments 2

2 Overall Description 3

2.1 Product Perspective 3

2.2 Product Functionality 3

2.3 Users and Characteristics 3

2.4 Operating Environment 3

2.5 Design and Implementation Constraints 4

2.6 User Documentation 4

2.7 Assumptions and Dependencies 4

3 Specific Requirements 5

3.1 External Interface Requirements 5

3.2 Functional Requirements 6

3.3 Behaviour Requirements 6

4 Other Non-functional Requirements 7

4.1 Performance Requirements 7

4.2 Safety and Security Requirements 7

4.3 Software Quality Attributes 7

5 Other Requirements 8

Appendix A – Data Dictionary 9

Appendix B - Group Log 10

Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| Version 1, rev. 0.0 | Thomas Bailey  Henry Unruh | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 10/25/19 |

# 

# Introduction

Buff Stuff is an interactive web app designed to be used from a mobile device to manage lifestyle choices and improve user health.

## Document Purpose

The purpose of this document is to present a concept of a web application that is designed to promote healthier activities and manage user health. The systems involved for our product will involve some data. Given this, some of the software will involve data manipulation and storage. The scope will mostly cover user interaction and data collection, that the user will input in some form for the program to parse and digest and those interactions (data gathering and user interactivity) are subsystems of the main site which will house these components.

## Product Scope

Our product **Buff Stuff** is an interactive web application that is used to monitor and track the user’s lifestyle choices and fitness activities. It will allow users to create an account that manages their current habits. This allows us to gather data on their lifestyle and give the user the ability to generate and manage fitness goals.

With data collection, it allows us to compare and contrast elements such as caloric intake compared to calories burned to give the user a granular indicator of their progress and with additional information, the product can offer reasonable goals and ideas that the user can engage in to help improve their health. This achieves our vision to create a relatively basic and simple program to analyze fitness and promote better health.

## Intended Audience and Document Overview

The intended audience for this product are primarily for individuals looking to improve their fitness or to start goal-setting. The audience that is intended for this document could be marketing executives and potential developers. Marketing individuals can use this application to promote healthier behaviors for individuals that need an outline.

Users will be the individuals that download and interact with the application.

Developers can utilize this document and have a general understanding of what this program is trying to achieve and integrate other third-party software or hardware to the application.

## Definitions, Acronyms and Abbreviations

BS: Buff Stuff – our product.

SP: Swole Patrol – our team.

HTTPS: Hyper Transfer Text Protocol Secure – used for websites and to ascertain that the site is secure.

UI: User Interface.

## Document Conventions

This document follows standard IEEE standards for formatting, which includes the use of standardized fonts, font sizes, and other miscellaneous requirements. This section will be expanded upon as future SRS iterations are created.

## References and Acknowledgments

<insert references and acknowledgements at a later date.>

# Overall Description

## Product Perspective

**Buff Stuff** is a product designed to aid in making and maintaining healthier lifestyle choices. It is based around the idea that many fitness trackers allow for calorie counting and excercises performed along with other features but rarely incorperate them all into a single usable product. This systems design gives the user the ability to create a new plan or account to monitor their status or use existing third-party software for an existing plan.

Monitoring will utilize nutritional databases to calculate excess calories, calories in food, and other fitness actions. Optionally, if you’re extra spicy, you can also post your results to social media platforms like InstaGram or Facebook.

The status is tracked through using goals set by the user. Once a goal has been set, the user can then perform different lifestyle activities to see whether they meet their criteria or not after some given time. Our product will utilize various databases in regards to nutritional values, such as caloric intake and calories burned.

## Product Functionality

The main functions of the product are as follows:

* Track the total calories that have been burned/consumed.
* A progress tracker that provides the following:
  + Auto generated weekly reports of finess goals.
  + A daily goal of caloric intake.
  + Weekly calorie counter: burned vs. consumed.
  + Nutritional values and vitamins tracker.
* Has access to databases that:
  + Provide nutritional information of certain foods (like a banana).
  + Provides information about various exercises that can be performed.
* User-based goal-setting, with dates and plans.

## Users and Characteristics

The users of the product will be the general public – free of use for everyone, however the scope isn’t inherently limited to this, fitness trainers and gyms may also use this application.

Regular users are defined as people who seek to improve their health physically and wish to have a numerical journal to catalogue and share the progress of their journey. Fitness trainers and gyms can also utilize this app to help create goals for regular users as well as establish potential plans for other users.

## Operating Environment

The operating environment will be on a computer or mobile device(maybe). The software that is pertinent will be any operating system like Windows 7/10, macOS, iOS, Android, and Linux.

## Design and Implementation Constraints

You need access to free databases for exercises and nutritional content. The access of some social media platforms like account linking is also a possibility. In addition to this, developers will need access to the software details for third-party accessories like FitBit or other tracking applications. Hardware compatability is also a potential issue, since some of the guidelines for certain platforms are different and controlled by different manufacturers.

## User Documentation

If the user is confused about any particular element(s) of the application, the interface will have a small help site. This help site will help diagnose basic questions and inquiries. This documentation will contain information about setting up the weekly logs for fitness tracking, basic information about goal tracking as well as mobile push notifications. Other elements of the documentation could include an end-user license agreement for the use of the app moreso as a disclaimer, and for further explanation, a wiki page dedicated to answering and going through steps on how to perform certain steps.

## Assumptions and Dependencies

It will be assumed that the userbase will have accesss to some mobile device or computer that is easily or readily available. It is optional to have third-party devices like a FitBit, smart watch, or social media presence on sites like Facebook or Instagram, and of course, the willingness to improve one’s physical prowess!

# Specific Requirements

## External Interface Requirements

### User Interfaces

Manual input is going to be the main way that the user will engage with the software. There will be some basic graphics that will lead the user to inputting certain pieces of data, like calories burned, consumed, etc. Search functions for exercises the user would like to learn about and foods that they would want to investigate. A settings option for adjusting acocunt preferences and notifications will also be present.

### Hardware Interfaces

The hardware that will be associated will be things like a mobile device to carry the app, but will also be able to interface with other functions of the phone, such as the pedometer. Optionally, other tools can be linked together, like smart-watches (apple watch), FitBits et cetera.

### Software Interfaces

The software that will be responsible with linking to a database that will provide an updated page for the user’s nutritional values for the week. Additionally, there will be a second database for information about exercise techniques for the end-user to integrate within their routine. Included in the account settings and preferences help menu, it will have a link to the **Buff Stuff** website FAQs. Other outside networks that the app will be capable of communicating with will be the cloud for information storage as well as social media for sharing and updating progress on fitness goals.

### Communications Interfaces

The main website for providing services for the **BS** app will follow the HTTPS encryption standard on top of having both the site and app having two-factor authentication that allows for greater security and privacy.

## Functional Requirements

The application will have three main categories in the design of the program. In a susinct format:

The progress is the primary function of the **BS** app. It provides two secondary functions which are: a weekly progress report that provides the user with calories consumed vs burned and a overall caloric goal for that week. If the nutrition tracker portion of BS is used, it allows the total nutritional value of foods to be displayed for that week as well. Finally, the weekly report will also display the exercises the user has engaged with in a total body workout diagram.

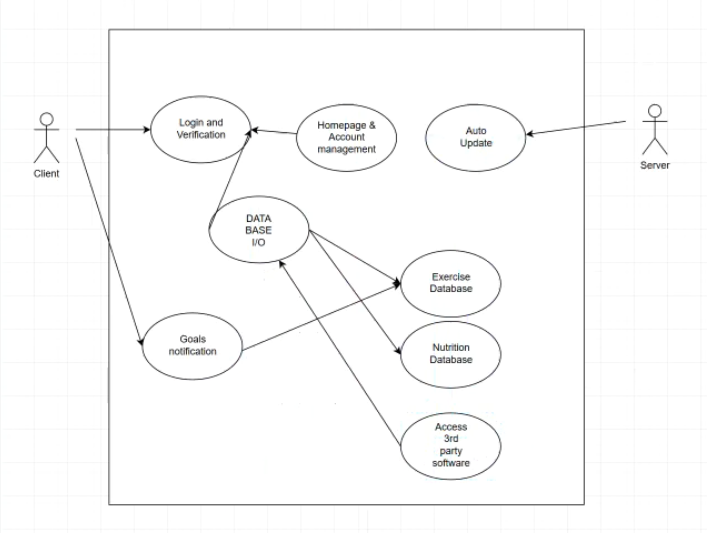
The other subfunction of the progress tracker is to set goals. With this function, the user can determine the timeline of their goals relating to their overall fitness and weight. The system will then attempt to generate a program for the user to follow to achieve their plan, including a diet plan. At the end of the plan, it will compare the initial entries with the current entries to give a before and after snapshot of the user’s fitness.

The second, primary function is to track the calories burned throughout the week. In order to accomplish this, it is capable of monitoring the duration and intensities of the exercises performed by the user through the mobile device’s pedometer and/or third-party accessories such as a FitBit or smart watch.

The final primary function is to record the total calories consumed; this function acts in conjunction with the calories burned function to provide the weekly tracker with updated information. It consists of a manual entry mode for the total number of exercises that the user has performed. It also contains a search database for new exercises for specfic muscle groups in the event that the user wants to vary their routine.

## Behaviour Requirements

### Use Case View



# Other Non-functional Requirements

## Performance Requirements

For our performance requirements, we intend to provide the following:

* The program should be capable of an offline mode, in the event that Wi-Fi isn’t available at their current location, this is a particularly useful for long-distance cycling or running.
* The boot time for the app should be less than ten seconds for the app load and less than 5 for the app to change pages.
* The application should receive automatic updates from a server to ensure stability.
* The application should be compatible and format with various mobile devices.
* User-friendly touch screen prompts.
* The ability to easily maniplate data, such as adding or removing, and the option able to delete your account.

## Safety and Security Requirements

The **Buff Stuff** application will take advantage of multiple security protocols to include: two-factor authentication, the potential for biometric security such as facial or fingerprint recognition, the supporting **BS** website utilizing HTTPS encryption, and finally enouraging the user to create a password that is at least greater than eight characters and three different symbol types to include an uppercase letter, a number, and a special character.

## Software Quality Attributes

During updates, the databases will be refreshed with the latest information regarding nutritional values and exercises as well as providing user-end support and security. After the user inputs new information into the application, user inputted data and account data should be reflected within the system immediately. The program will have stable access to the cloud and being able to store user data as well as have access to information databases. The UI experience should be friendly and easy to understand.

# Other Requirements

N/A.

Appendix A – Data Dictionary

* Tracks calories burned
  + Access fit bit
  + Manual entry mode
  + Pedometer function for phone
* Tracks calories consumed
  + Manual entry mode
  + Food nutrition database
* Progress tracker
  + Weekly progress report
    - Consumed vs burned
    - Nutrition tracker
    - Exercise tracker

Appendix B - Group Log

<October 18th> Bailey generated SRS document draft. Generated outline for program. Uploaded to github.

<October 20th-21st> Group collaboration on more ideas of draft and product features. Began skeleton draft for said idea and filled in basic ideas for product. Henry and Bailey added rough ideas to Introduction (1.1, 1.2, 1.3) and Overall Description (2.2).

<October 22nd-23rd> Henry added details to Introduction (1.1, 1.2, 1.3) and Overall Description (2.2).

<October 24th> Henry added details to Introduction (1.1, 1.2, 1.3) and Overall Description (2.2). Uploaded to github. Bailey added details to Non-functional Requirements (4.2). Bailey created skeleton for software requirements. SP team brainstormed ideas for feature to be included.

<October 25th> Henry and Bailey polished the SRS. Turned in for pending grade.