Software Requirements Specification

for

Fit This!

Version 1.0 approved

Prepared by Ryan Techmeier, Jenny Druckrey, Paul Loehlein, Joe Wennesheimer, Jake Rose

Northeast Wisconsin Technical College

9/11/2018

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Remove User Classes	11/26/18	Not enough time to incorporate	1.1
Remove Fat Secret	12/4/18	Not enough time to incorporate	1.2

1. Introduction

1.1 Purpose

Fit This (Version 1.0) is an application to track personal fitness activities and details.

- It will contain health related calculations such as:
  BMR - basal metabolic rate (number of calories you burn when your body is at rest)
  BMI - body mass index (a number calculated from your height and weight which is then used to assess your body composition)
  Weight change
- Allow users to track:
  Caloric intake by recording meal details
  Calories burned via recorded fitness activities
  Weight
- Allow individual user accounts
  User information will be kept separate from other user information

1.2 References

Database Application - SQLite https://www.sqlite.org/index.html

2. Overall Description

2.1 User Classes and Characteristics

There is one user class. The user class is free. The user will be able to set up an account. Track weight change, caloric intake through specific types of food consumption records, and calories burned through specific types of exercise. The values derived from these records will calculate and display weight change, BMI, BMR, and suggested daily caloric intake to reach the user's desired target weight.

2.2 Operating Environment

Microsoft Win32 compatible OS with .NET Framework 4.6.1, 1 core or more, 4 GB Ram (Mainly to support a smooth running operating system), 5 Mb free storage.

2.3 Design and Implementation Constraints

This development cycle is limited by time and workforce availability. Therefore, our ability to produce features depends on the time available up to the course due date, and the ability of

each student member to place effort towards development. Some student members may choose to put more or less effort into this project depending on their other life / class obligations.

2.4 Assumptions and Dependencies

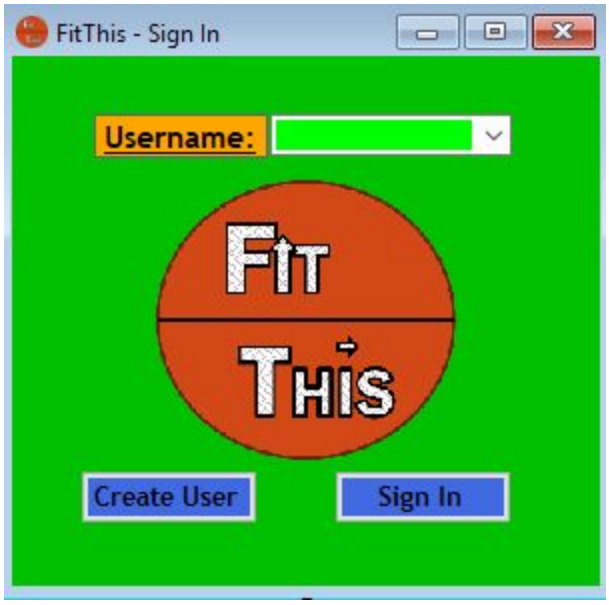
The Sqlite database package will be incorporated into the project without issues, and supporting all feature sets we choose to incorporate

Windows Forms will allow us to create a graphical user interface with text fields, graphs, and tables to display user guidance, critical data, and recorded values.

3. External Interface Requirements

3.1 User Interfaces

Login Screen



Most of the user interfaces will be similar to the form show in the image above. The application will use standard Windows Form controls with minor customization. The application background will be either gray if it is a low profile screen like the “Create User Screen” which is shown in the image below, or green if it will be a common use screen like the tabbed dashboard, which the user sees once they log in.

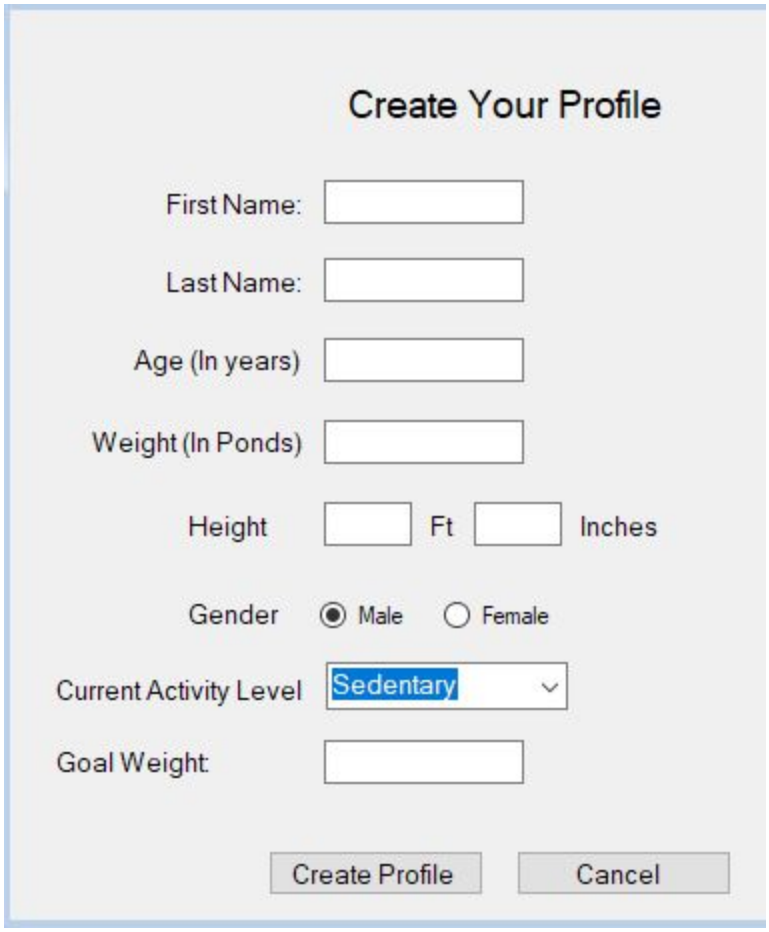
The buttons will all have the standard view aside from those seen on the sign in form shown above. Important buttons will have the text bold and or underlined.

Page headings will be written in bold and a larger font than the standard font used for labels and subsections.

Display information will be shown on a standard white textbox for ease of use.

All unmentioned interfaces will be standard, as to not draw extra attention.

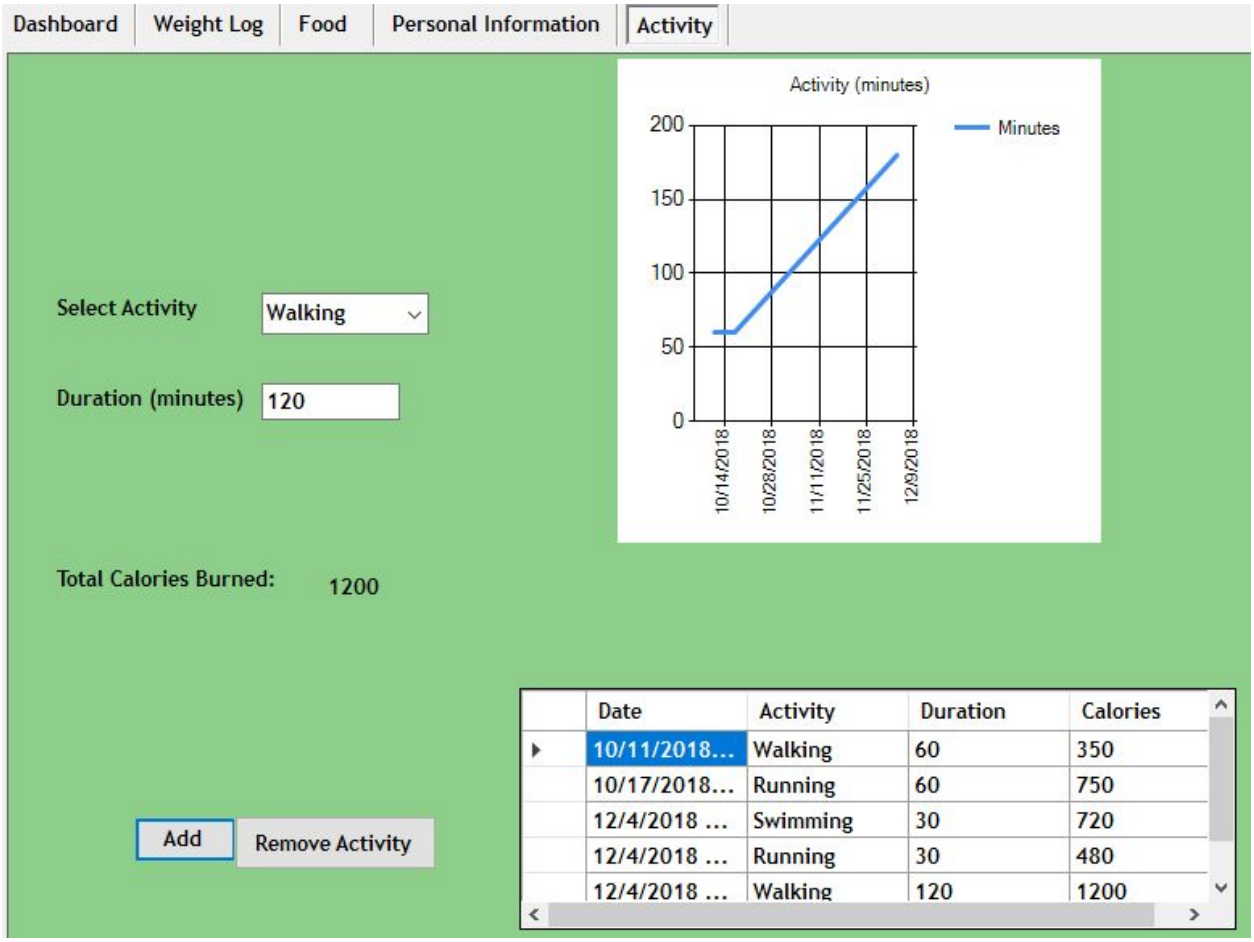
Create User Screen



The display resolution for all pages will fit within an area of 1000 pixels wide and 700 pixels tall.

Fonts will be mostly Trebuchet MS.

Graphs and tables will follow the format seen in the image below. The lines may be red or blue depending on the context and page scheme.



Hardware Interfaces

The hardware for this application will be the standard keyboard and mouse. A touch screen may be used for the same selection options that the mouse would provide.

Data will be entered with a physical keyboard, or on-screen operating system touchscreen keyboard. Information will be viewed with any PC display compatible with the system which is greater than or equal to 1024x768.

3.2 Software Interfaces

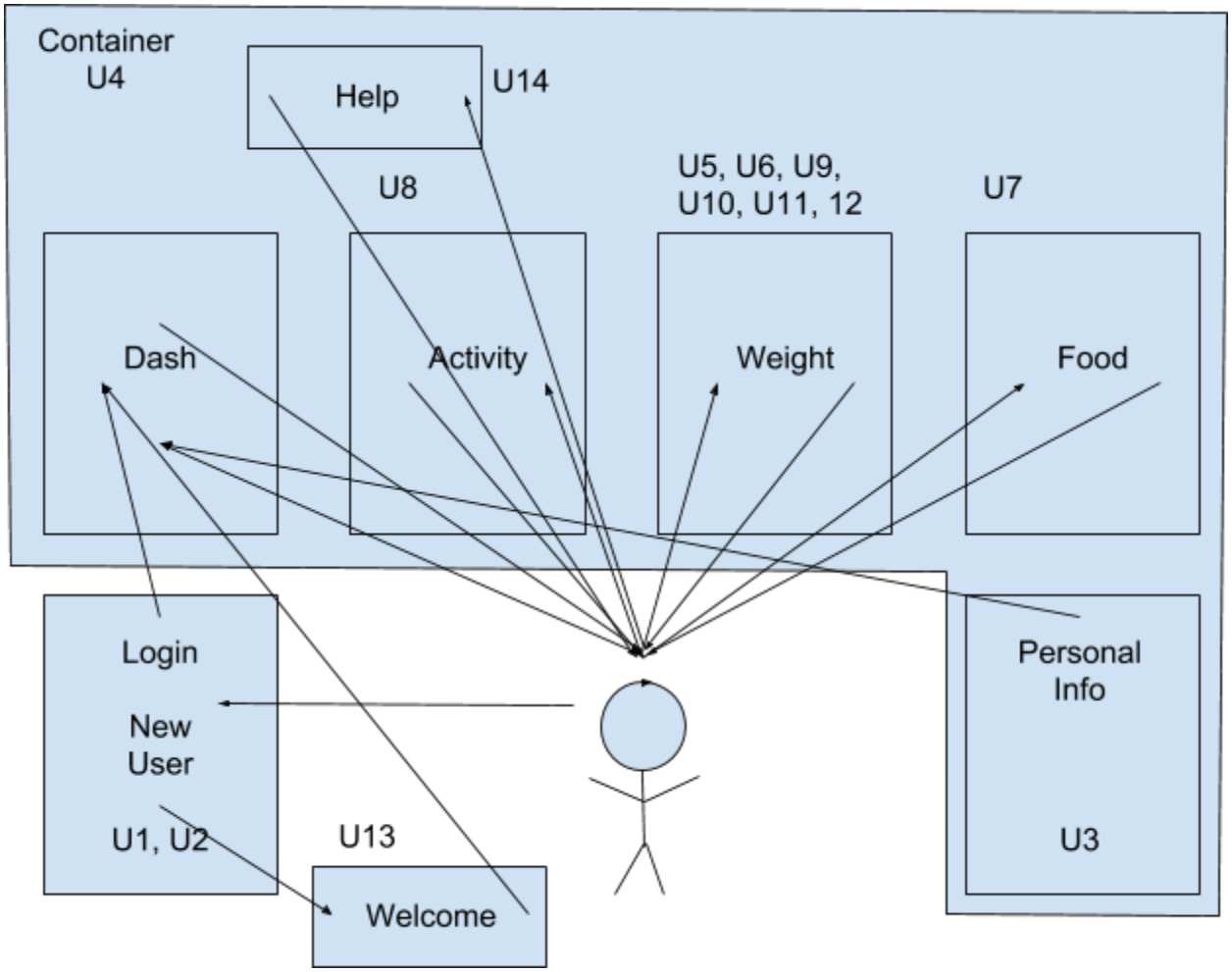
This application will interface with the standard Windows 32-bit compatible operating system with the .NET 4.6.1 Framework installed. It will use the Sqlite database which comes with a .NET interface with the application uses to write data to and read data from a database file.

3.3 Communications Interfaces

This application contains no external communication interfaces. There is an internal Sqlite plugin that has a communication string to access the database file.

4. System Use Cases

4.1 Use Case Diagram



4.1 Create User (U1)

1. **U1**
2. **Description** – A first-time user creates a new user account.
3. **Priority** – High

4. **Trigger** –User selects "Create new User" button. (External)
5. **Actors** – User
6. **Flow of Events**
   6.1. **Basic Flow**
      6.1.1. User enters necessary information to create and account: Name starting weight, height, activity level.
      6.1.2. User information is stored in an internal database associated with their user account.
   6.2. **Alternative Flow**
      6.2.1. User canceled out of create new user form and is returned to the login form
   6.3. **Exception Flow**
      6.3.1. An error message dialog is displayed informing the user of what needs to be corrected before the Create user will be accepted
7. **Preconditions** – User has selected "Create new user" from the login form
8. **Post conditions** – A new user is created. Their information is available in the program database for reference & calculations. The new user's dashboard is displayed.
9. **Notes/Issues** - None

4.2 Load User (U2)

1. **U2**
2. **Description** – An existing user logs into their account.
3. **Priority** – High
4. **Trigger** – User presses the Login button after selecting the username (External)
5. **Actors** – User
6. **Flow of Events**

6.1 **Basic Flow**

6.1.1 User selects their user name from a drop-down menu.

6.1.1.1 Current user names are displayed in the dropdown menu

6.1.2 User clicks button to log into the program

6.1.2.1 User information is loaded from the internal database

6.2 **Alternative Flow**

6.2.1 User closes the program

6.3 **Exception Flow**

6.3.1 An error message dialog is displayed informing the user of what needs to be corrected before the login information will be accepted.

7. **Preconditions** – Startup/Login screen is displayed
8. **Post conditions** – The user's dashboard, displaying current statistics, is displayed. User can access other program tabs.
9. **Notes/Issues** - None

4.3 Display User Information (U3)

1. **U3**
2. **Description** – User will go to the page and that page will display the user's personal info

3. **Priority** – Medium
4. **Trigger** – Navigate to screen or may be first screen visible after login (External)
5. **Actors** – User
6. **Flow of Events**

6.1 **Basic Flow**

6.1.1 On page load the user will see the items that were entered in the "create view" sections, other than the password if a password is needed.

6.1.1.1 The page load would trigger the retrieval of data from the user class and display that data on a form for the user.

6.2 **Alternative Flow**

6.2.1 N/A

6.3 **Exception Flow**

6.3.1 N/A

7. **Preconditions** – User will have to be create and logged in. User will have to take the actions to view the page containing the personal information(login home page, or click to view)
8. **Post conditions** – The user's dashboard, displaying current statistics, is displayed. User can access other program tabs.
9. **Notes/Issues** - None

4.4 Log User Out (U4)

1. **U4**
2. **Description** – User is logged in. User dashboard & navigation systems are visible
3. **Priority** – Medium
4. **Trigger** – User selects logout icon, or close program (External)
5. **Actors** – User
6. **Flow of Events**

6.1 **Basic Flow**

6.1.1 User selects logout icon from the screen

6.1.2 User is logged out

6.1.3 All user information is saved in database

6.2 **Alternative Flow**

6.2.1 N/A

6.3 **Exception Flow**

6.3.1 N/A

7. **Preconditions** – User will have to be created, User will have to be logged in.
8. **Post conditions** – The user's dashboard, displaying current statistics, is displayed. User can access other program tabs.
9. **Notes/Issues** - None

4.5 Display BMI (U5)

1. **U5**
2. **Description** – Calculates the user's Body-Mass Index

3. **Priority** – Medium
4. **Trigger** – Calculate after button click and display BMI (External)
5. **Actors** – User
6. **Flow of Events**
   - 6.1 **Basic Flow**
     - 6.1.1 User enters height and weigh on information form
     - 6.1.2 User clicks button to display BMI
     - 6.1.3 Information is retrieved from the User DB
     - 6.1.4 BMI is displayed
   - 6.2 **Alternative Flow**
     - 6.2.1 N/A
   - 6.3 **Exception Flow**
     - 6.3.1 N/A
7. **Preconditions** – User navigates to the tab with BMI listed.
8. **Post conditions** – User has the option to navigate elsewhere.
9. **Notes/Issues** - None

4.6 Display BMR (U6)

1. **U6**
2. **Description** – Calculates the user's Basal Metabolic Rate
3. **Priority** – Medium
4. **Trigger** – Button click (External)
5. **Actors** – User
6. **Flow of Events**
   - 6.1 **Basic Flow**
     - 6.1.1 User enters height and weigh on information form
     - 6.1.2 User clicks button to display BMR
     - 6.1.3 Information is retrieved from the User DB
     - 6.1.4 BMR is displayed
   - 6.2 **Alternative Flow**
     - 6.2.1 N/A
   - 6.3 **Exception Flow**
     - 6.3.1 N/A
7. **Preconditions** – Form is available.
8. **Post conditions** –Weight is logged into database.
9. **Notes/Issues** - None

4.7 Display Caloric Intake (U7)

1. **U7**
2. **Description** – Displays the number of calories the user has eaten
3. **Priority** – High
4. **Trigger** – Calories calculated after button click (External)
5. **Actors** – User

6. **Flow of Events**
   6.1 **Basic Flow**
      6.1.1 User enters Calories
      6.1.2 User clicks button to display total Calories
      6.1.3 Calories are totalled and displayed
   6.2 **Alternative Flow**
      6.2.1 N/A
   6.3 **Exception Flow**
      6.3.1 N/A
7. **Preconditions** – User must have number of calories entered
8. **Post conditions** – User has the option to navigate elsewhere.
9. **Notes/Issues** - None

4.8 Display Burned Calories (U8)

1. **U8**
2. **Description** – Displays the number of calories the user has burned off from exercise
3. **Priority** – High
4. **Trigger** – Calculated after button click (External)
5. **Actors** – User
6. **Flow of Events**
   6.1 **Basic Flow**
      6.1.1 User enters exercise and duration
      6.1.2 User clicks button
      6.1.3 Previous information is retrieved from the database to total calories burned
      6.1.4 Calories burned is displayed
   6.2 **Alternative Flow**
      6.2.1 N/A
   6.3 **Exception Flow**
      6.3.1 User can choose not to log exercise (did not exercise)
7. **Preconditions** – Calories burned is logged into the database
8. **Post conditions** – User has the option to navigate elsewhere.
9. **Notes/Issues** - None

4.9 Goal Weight (U9)

1. **U9**
2. **Description** – Tracks the user's weight, and compares to the user's goal weight.
3. **Priority** – Medium
4. **Trigger** – User enters a goal weight (External)
5. **Actors** – User
6. **Flow of Events**
   6.1 **Basic Flow**
      6.1.1 User enters in a desired weight at sign-up
      6.1.2 Weight is saved in the internal database

6.1.3 Goal weight is added to graph as note

6.2 **Alternative Flow**

6.2.1 Program determines if user is trying to gain or lose weight, and sets calorie goal accordingly.

6.3 **Exception Flow**

6.3.1 N/A

7. **Preconditions** – User has entered at least one weight log
8. **Post conditions** – User has entered and saved goal weight and sees is graph updated
9. **Notes/Issues** - None

4.10 Record Weight (U10)

1. **U10**
2. **Description** – Users weight history is saved after entry by user.
3. **Priority** – Medium
4. **Trigger** – Button click (External)
5. **Actors** – User
6. **Flow of Events**

6.1 **Basic Flow**

6.1.1 User enters current weight

6.1.2 Weight and date of entry is saved to local database

6.1.3 Weight and date of entry are saved in the database

6.2 **Alternative Flow**

6.2.1 Possible option to view weight history as table and user can adjust weight history or delete logs

6.3 **Exception Flow**

6.3.1 N/A

7. **Preconditions** – User has entered a weight log
8. **Post conditions** – User weight log and date of entry are saved
9. **Notes/Issues** - None

4.11 Display weight history (U11)

1. **U11**
2. **Description** – Users weight history is displayed as a line graph
3. **Priority** – Medium
4. **Trigger** – Table and graph displayed after button click (External)
5. **Actors** – User
6. **Flow of Events**

6.1 **Basic Flow**

6.1.1 Weight is entered and button is clicked

6.1.2 Weight and current data are saved to database

6.1.3 Weight history and dates of entry history are pulled from the local database and are graphed along with the most recent entry.

6.1.4 Graph is displayed

6.2 **Alternative Flow**
6.2.1 Weight goal is met message displays congratulating user. This happens if the user went below the desired weight when goal was to lose, or if weight went above goal when user was trying to gain then the message is displayed.
6.3 **Exception Flow**
6.3.1 N/A
7. **Preconditions** – At least one weight is saved in weight data table.
8. **Post conditions** – Users weight log was saved and the graph was adjusted for most recent value entered.
9. **Notes/Issues** - None

4.12 Log Meal, Display Log Table, Display Log Graph (U12)

1. **U12**
2. **Description** – Logs a meal, Displays/Updates table and graph for meal
3. **Priority** – High
4. **Trigger** – (External)
5. **Actors** – User
6. **Flow of Events**
6.1 **Basic Flow**
6.1.1 Enter title of meal
6.1.2 Enter Calories of meal
6.1.3 Press Save button
6.2 **Alternative Flow**
6.2.1 N/A
6.3 **Exception Flow**
6.3.1 N/A
7. **Preconditions** – User is on the food log page and the food log database is available.
8. **Post conditions** – Food Log saves another record, meal table and graph are updated
9. **Notes/Issues** - None

4.13 Welcome Window (U13)

1. **U13**
2. **Description** – Window will show a welcome message for new user accounts
3. **Priority** – Medium
4. **Trigger** – (External)
5. **Actors** – User
6. **Flow of Events**
6.1 **Basic Flow**
6.1.1 User clicks button to complete sign up process
6.1.2 Welcome window is displayed
6.1.3 User closes welcome window
6.2 **Alternative Flow**
6.2.1 N/A

6.3 **Exception Flow**
6.3.1 N/A
7. **Preconditions** – User must correctly complete the new account form to sign up prior to clicking the okay button
8. **Post conditions** – User will be taken to the dashboard screen
9. **Notes/Issues** - None

4.14 Help Menu (U14)

1. **U14**
2. **Description** – User will see a window describing instructions on how to use the application
3. **Priority** – Medium
4. **Trigger** – (External)
5. **Actors** – User
6. **Flow of Events**
6.1 **Basic Flow**
6.1.1 User will click on the Help button
6.1.2 The help window will be displayed with a list of instructions for use
6.1.3 User will have the option to close the window
6.2 **Alternative Flow**
6.2.1 N/A
6.3 **Exception Flow**
6.3.1 N/A
7. **Preconditions** – User must be logged into the application before the help button is displayed
8. **Post conditions** – After help has been closed, user is taken back to the previous screen they were on prior to selecting help
9. **Notes/Issues** - None

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Joe's laptop does not have the proper performance requirement. Do not use Joe's laptop to run this application. Actually, Using Joe's laptop would be a Nonfunctional Requirement...

5.2 Safety Requirements

This application is designed for entertainment purposes only. No health related decisions should be made based on the information or suggestions offered. Any and all health related decisions should be made in conjunction with a health field expert trained to offer advice in this field.

5.3 Security Requirements

There are no security features built into this application. It is based on the honor system.

5.4 Software Quality Attributes

There will be no software quality attributes build into the application.

6. Other Requirements

None

7. System Requirements Chart

ID	Priority	Type	Source	Contain UC	Description
1	High	F	Jenny	UC1	Create User
2	High	F	Jenny	UC2	Load User
3	Medium	F	Paul	UC3	Display User Information
4	Medium	F	Jenny	UC4	Log User Out
5	Medium	F	Paul	UC5	Display BMI
6	Medium	F	Paul	UC6	Display BMR
7	High	F	Ryan	UC7	Display Caloric Intake
8	High	F	Jake	UC8	Display Burned Calories
9	Medium	F	Joe	UC9	Goal Weight
10	Medium	F	Joe	UC10	Recorded Weight
11	Medium	F	Joe	UC11	Display Weight History
12	High	F	Ryan	UC12	Log Meal, Display Log Table & Graph
13	Medium	F	Jenny	UC13	Welcome Window

14	Medium	F	Jenny	UC14	Help Menu