

# HerdHealth: A Mobile Health Data Aggregator for the WPI Community

A Major Qualifying Project

Submitted to the Faculty of  
Worcester Polytechnic Institute  
In partial fulfillment of the requirements for the  
Degree in Bachelor of Science  
in  
Computer Science

By:

---

Adam Bettigole

---

Tessa Garbely

---

Taehyun Kim

---

Jessie Ying

---

Professor Emmanuel O. Agu, Advisor

---

Professor Craig E. Wills, Department Head

## **Abstract**

While many health and fitness smartphone applications exist, not all applications support all data types, which results in fragmented data and reduces the efficacy of Behavior Change Techniques (BCTs). The purpose of this MQP was to create an application that allows members of the WPI community to aggregate all of their data to get a complete view of their health progress. Consolidating all of one's health data can help minimize health-related risks, find trends over time, and increase user engagement. Our app includes features that implement several behavioral change mechanisms to help users achieve their health goals. While beta testers found the app useful overall, we also found areas for improvement. These potential improvements and our ideas for additional features could serve as the basis for a future project.

## **Acknowledgements**

First, we would like to thank our advisor, Professor Emmanuel Agu, for inspiring the idea and assisting us throughout the entire project.

We would also like to thank Ruth McKeogh, who represented the Institutional Review Board at Worcester Polytechnic Institute and helped us through the research application process. In addition, thank you to Jeanine Skorinko for helping us post our initial survey on the SONA Systems platform. Because of her, students who completed the survey received proper credit, and we obtained the results we needed.

Finally, thank you to everyone who participated in both our initial survey and our evaluation sessions. This project could not have been completed without your input.

## List of Figures

Figure 1: Predicted Growth of Wearable Technology .....	4
Figure 2: Illustrated Vision of HerdHealth .....	10
Figure 3: Number of Days Intervention was Used by Participants in Each Group .....	16
Figure 4: Diagram of Structure of Siloed (Stovepipe) Approach vs. Open mHealth .....	21
Figure 5: Diagram of Fitness APIs and Data They Provide .....	23
Figure 6: Statuses (Student, Faculty/Staff, or Other) of Initial Survey Respondents, N=251 .....	27
Figure 7: Gender Distribution of Initial Respondents, N=251 .....	28
Figure 8: Ages of Initial Survey Respondents, N=250 .....	29
Figure 9: Activity Levels of Initial Respondents, N=251 .....	29
Figure 10: Types of Mobile Phone Owned by Initial Respondents, N=251 .....	30
Figure 11: Respondents Who Use Application/Device, N=261 .....	31
Figure 12: Do You Use Social Features in Your Health Apps? .....	32
Figure 13: Do You Feel Using These Social Features Has Helped You Improve Your Health?. .....	32
Figure 14: Data Types Tracked by Respondents, N=202 .....	33
Figure 15: Respondents with Desire to Improve Areas, N=226 .....	34
Figure 16: Respondents Who Want to Compare Data Types with Friends, N=158 .....	35
Figure 17: Level of Interest in Application, N=257 .....	36
Figure 18: Summary of Initial Survey Results .....	38
Figure 19: System Architecture Diagram .....	39
Figure 20: Workflow of Application (Part 1) .....	42
Figure 21: Workflow of Application (Part 2) .....	43
Figure 22: Workflow of Application (Part 3) .....	44
Figure 23: Implementation Diagram.....	49
Figure 24: Diagram of Database in Firebase .....	53
Figure 25: Sample JSON Response .....	56
Figure 26: Sample Java Class .....	56
Figure 27: Diagram of Classes Saved in Realm .....	61
Figure 28: Custom Classes Not Implemented in Realm .....	65
Figure 29: IDataTypes Interface Implementation .....	67
Figure 30: ITaskFinished Interface Implementation.....	67
Figure 31: WPI Status of Evaluation Survey Respondents .....	71
Figure 32: Gender of Evaluation Survey Respondents .....	72
Figure 33: Ages of Evaluation Survey Respondents .....	72
Figure 34: Activity Levels of Evaluation Survey Respondents .....	73
Figure 35: Phone Type Owned by Evaluation Survey Respondents .....	74
Figure 36: Screenshot of Statistics Screen.....	78
Figure 37: Screenshot of Home Screen .....	78
Figure 38: Screenshot of Profile Page .....	79
Figure 39: Average Likelihood to Use HerdHealth Frequently, by Activity Level .....	81

## List of Tables

Table 1: Highest Ranked Health and Fitness Applications on Android .....	6
Table 2: A Sample of Applications and the Types of Data They Track.....	8
Table 3: Details of BCTs We Plan to Use .....	12
Table 4: Tools Used in HerdHealth and How They Were Applied.....	49
Table 5: Applications Supported by GetHealth and Associated Data Types .....	50

## Table of Contents

1. Introduction.....	1
1.1 Importance of Exercise.....	1
1.2 Growing Use of Health and Fitness Wearable Technology.....	3
1.3 Growing Use of Health and Fitness Applications.....	4
1.4 Problem.....	6
1.4.1 Use Case 1.....	6
1.4.2 Use Case 2.....	7
1.5 The Goal of This Major Qualifying Project (MQP).....	9
1.6 Project Vision.....	9
2. Literature Review.....	14
2.1 Studies on Effectiveness of Technology-Based Interventions.....	14
2.2 Impact of Social Comparison and Competition.....	17
2.3 Community Features of Fitness Apps.....	18
2.4 Open mHealth.....	19
2.4.1 Open mHealth Vision.....	19
2.5 Our approach.....	22
3. Survey.....	24
3.1 Survey Results.....	26
3.1.1 Demographics.....	26
3.1.2 Apps and Devices Used by Respondents.....	30
3.1.3 Social Features.....	31
3.1.4 Types of Data Tracked.....	33
3.1.5 Interest in HerdHealth.....	35
4. HerdHealth System Architecture.....	39
4.1 Back End.....	40
4.1.1 Retrieving Data.....	40
4.1.2 Cloud Database.....	40
4.1.3 Local Database.....	40
4.2 Front End.....	41
4.2.1 Login and Sign Up.....	44
4.2.2 Home Page.....	45

4.2.3 Statistics Page.....	45
4.2.4 Feed Page.....	46
4.2.5 Settings Page.....	46
5. Implementation .....	48
5.1 Android Application Design .....	48
5.2 Tools.....	48
5.2.1 GetHealth.....	49
5.2.2 Google Firebase .....	52
5.2.3 Realm.....	54
5.2.4 OkHttp .....	55
5.2.5 Gson.....	55
5.2.6 GraphView.....	56
5.2.7 Glide .....	56
5.3 Development Decisions.....	57
5.3.1 GetHealth Sync Google Cloud Function .....	57
5.3.2 Selectively Retrieving GetHealth Data from Application .....	58
5.3.3 Nutrition Goal Achievement Streaks.....	59
5.4 Custom Java Classes and Interfaces.....	60
5.4.1 Custom Classes in Realm .....	61
5.4.2 Custom Classes Not in Realm .....	64
5.4.3 Custom Interfaces .....	67
6. Evaluation .....	70
7. Results.....	71
7.1 Evaluation Survey Results .....	71
7.1.1 Demographics.....	71
7.1.2 System Usability Scale .....	74
7.1.3 Most Liked Feature.....	74
7.1.4 Least Liked Feature .....	75
7.1.5 Additional Types of Data Users Want to Track .....	75
7.1.6 Suggestions for Application’s Appearance .....	75
7.1.7 Features Users Want in the Future .....	76
7.2 Observations from Beta Testing.....	77

8. Discussion .....	80
8.1 Discussion of Evaluation Survey Results.....	80
8.1.1 Evaluation Survey Findings.....	80
8.1.2 Potential Evaluation Survey Flaws.....	82
8.2 Potential Usability Improvements.....	82
8.3 Limitations of Evaluation.....	84
9. Conclusion .....	85
10. Future Work .....	86
10.1 Support More APIs.....	86
10.2 iPhone Version .....	86
10.3 Data Flexibility.....	87
10.4 User Groups.....	87
10.5 Statistics Filtering and Data Display.....	88
10.6 Friends and Feed Page.....	89
10.7 Challenges and Other Incentives.....	90
10.8 More Communities.....	90
References .....	92
Appendices.....	97
Appendix 1: Fitness App Survey .....	97
Appendix 2: Evaluation Survey .....	102
Appendix 3: System Usability Scale Results and Scoring Details.....	104



## **1. Introduction**

Over the past couple decades, technology has allowed people to automate many processes that used to be done manually, such as storing information digitally instead of on paper. In particular, this reform has greatly impacted how people track their health and fitness progress. Instead of keeping a food journal to track dietary habits, people can now keep track of everything they eat using smartphone applications, such as MyFitnessPal. Pedometers help people monitor their daily step counts, but they did not originally store previous data to help the user view and track progress. The introduction of wearable devices, such as Fitbits and Apple Watches, solved this issue by storing exercise data digitally and syncing it with the corresponding mobile applications and websites, granting people access to their exercise data on all their devices.

### **1.1 Importance of Exercise**

It has long been known that exercising consistently plays a vital role in keeping the human body healthy and strong. For example, exercise trained the heart to beat slower and stronger, so it would require less oxygen to function effectively. Physical activity also allowed for better blood flow because it helped arteries work more efficiently. In addition, exercise kept blood sugar levels down and high-density lipoprotein (HDL), also known as “good” cholesterol, up. Some studies have even suggested that exercise could reduce the chance that certain cancers, such as breast and colon cancer, return after remission (“Why we should exercise - and why we don't,” 2008). Several studies have also shown that exercise could affect the brain in positive ways, such as delaying or reducing the effects of depression and dementia (“Why we should exercise - and why we don't,” 2008).

Maintaining a healthy lifestyle was found to significantly reduce medical expenses. In fact, 50-75% of the \$2.7 trillion Americans spend annually on healthcare was related to preventable conditions associated with everyday behaviors, such as under-exercising, overeating, and smoking (Schüll, 2016). One study compared the average annual healthcare expenditures among people of varying levels of physical activity. The researchers gathered demographic, biometric, and exercise data from 51,165 adults, ages 21 years and older, and excluded those who were pregnant or reported difficulty or an inability to exercise. Respondents were classified into one of three categories: active (engaging in an average of at least 150 minutes of moderate-intensity activity per week), inactive (engaging in no physical activity), or insufficiently active (engaging in some physical activity, but not enough to be considered active). 45.5% of adults reported being active, while 34.2% were inactive, and 20.2% were insufficiently active. For each activity level, the researchers found the mean annual healthcare expenditure based on survey data. They found that those who reported being inactive spent an average of \$1,313 more per year than those who reported being active. The insufficiently active individuals reported spending an average of \$576 more per year than their active counterparts (Carlson, Fulton, Pratt, Yang, & Adams, 2015).

The benefits of engaging in physical activity on a consistent basis are clear, but many Americans still are not as active as is recommended by experts. The U.S. Department of Health and Human Services reported in 2010 that less than 5% of adults reached 30 minutes of physical activity every day, and only one in three met the recommended amount of activity per week (“President’s Council on Sports, Fitness & Nutrition,” 2012). One study found that one quarter of adults devoted none of their free time to exercise (“Why we should exercise - and why we don't,” 2008). The Active Healthy Kids Global Alliance is a network of researchers that

examines youth fitness levels of countries using nine different indicators. The United States, Australia, Canada, and England all received a “D-” for the physical activity of their youths (Tremblay et al., 2014). Given these statistics, it is clear that most people struggle to stay motivated or find the time to stay active, and it is affecting children too.

## **1.2 Growing Use of Health and Fitness Wearable Technology**

A 2013 report issued by the Pew Research Center stated that 70% of adults actively tracked their weight, diet, and exercise, and one in three adults tracked other health indicators, such as blood pressure, blood sugar, and sleep patterns (Schüll, 2016). However, at the time, only 20% of those who tracked their health patterns were using technology to do so (Schüll, 2016). Since then, the wearable technology market has seen resounding growth. Just a few years later, predictions are being made that wearable technology will experience yearly growth of 23% to over \$100 billion in 2023 and over \$150 billion in 2026 (Wade, 2017). Figure 1 shows how the revenue generated from wearable technology is predicted to grow through 2026. One key reason for such rampant growth is that wearable fitness trackers are not just for the young and healthy, as some thought when they first were introduced. In 2016, 17% of Americans over the age of 65 used wearables to track either their fitness or vital signs, compared to the 20% of Americans under 65 (Japsen, 2016). Whenever technology companies can appeal, not only to the young, tech-savvy population, but also to those over 65, there is potential for significant growth.

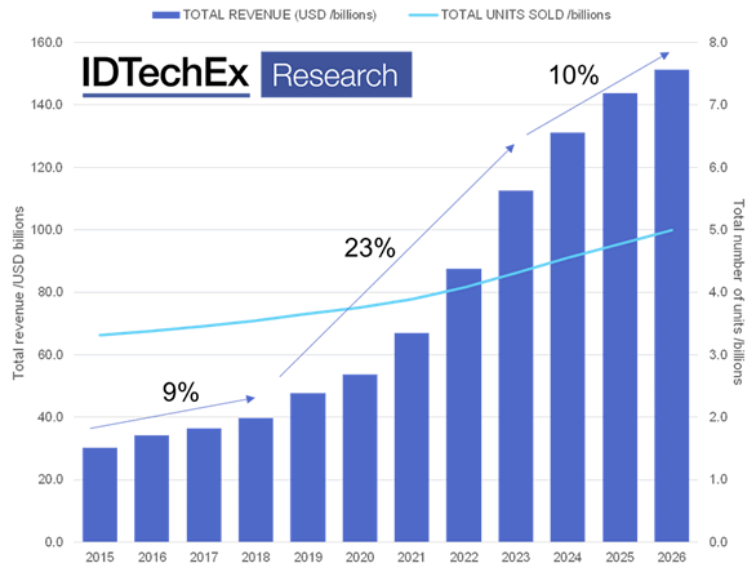


Figure 1: Predicted Growth of Wearable Technology (Wade, 2017)

### 1.3 Growing Use of Health and Fitness Applications

Smartphone applications, such as MyFitnessPal, can also be used to track health information, such as exercise and food intake. Applications often use the smartphone’s built in features, such as its accelerometer (used to track movement) and Global Positioning System (GPS), to track the user’s physical activity. Some can synchronize wirelessly with a wearable device as well, such as a Fitbit. Typical functions of health and fitness smartphone applications include tracking calories burned, sleep, and weight loss (Higgins, 2016).

Usage of health and fitness applications has increased dramatically in recent years; based on research of over 1 million applications (including, but not limited to fitness apps), fitness app usage on iPhone and Android increased more than 330% between 2014 and 2017 (Kesiraju & Vogels, 2017). Within the realm of fitness applications, the most commonly used type, by far, is the workout and weight loss app, which accounts for 73% of all fitness app usage, ranking above

general health, nutrition, and applications linked to fitness studios and gyms (Kesiraju & Vogels, 2017).

Table 1 shows information on some of the most installed health and fitness applications on Android devices as of October 6, 2018. These health and fitness applications are listed in the Google Play Store's top 500 free applications. The most installed application is Samsung Health, with over 500 million installations. The team hypothesizes that the large number of installations for Samsung Health and Galaxy Wearable apps is due to the fact that those apps come pre-installed on Samsung smartphones and smartwatches. However, the number of installs does not necessarily correlate with the applications' Google Play Store rankings. Rankings are based on several factors in addition to number of downloads, such as monthly active users, user ratings, and reviews (Ahn, 2017). This is likely why Samsung Health has a significantly lower ranking than Step Tracker, despite having far more installations. Even after taking into consideration the fact that the number of Samsung Health and Galaxy Wearable installations might not be indicative of how many people consistently use the applications, Table 1 still paints a picture of millions of individuals interested in using their mobile devices to improve their health.

Application	Number of Installations	Google Play Store Ranking
Samsung Health	500,000,000+	259
Galaxy Wearable (Samsung Gear)	100,000,000+	253
Calorie Counter - MyFitnessPal	50,000,000+	374
Fitbit	10,000,000+	389
Step Tracker - Pedometer, Daily Walking Tracker	5,000,000+	78
Fat Burning Workout - Home Weight Loss	1,000,000+	334
My Fitness Day - lose weight at home	100,000+	366

Table 1: Highest Ranked Health and Fitness Applications on Android

Adapted from (Apple Inc., 2019; Google, 2018)

## 1.4 Problem

Health applications and devices can be very useful in helping people to improve their diet and exercise habits. The problem is that not all applications support the ability to track all the health indicators that a user might want to monitor. This can lead to health data becoming fragmented between multiple applications and databases.

### 1.4.1 Use Case 1

If John wants to track exercise data, one option would be to use a Fitbit to keep track of daily steps taken, calories burned, and more. However, if John also wants to track dietary habits, he might turn to MyFitnessPal, a common choice for users who want to monitor their intakes of calories and macronutrients (protein, carbohydrates, and fats). John might also want to track

body weight and body mass index (BMI) using an iHealth smart scale. Now John has to open three different applications to view all of his health data.

#### *1.4.2 Use Case 2*

Another problem related to tracking health data is a lack of opportunities for social comparison. Many apps contain social features that allow users to form groups for comparing data and competing with friends, but using these features requires that all members of the group have the same application or device to track their progress. For example, if a member of the WPI women's basketball team wants to use the social features on MyFitnessPal to compare her dietary habits with those of her teammates, her teammates must also use MyFitnessPal, instead of MyPlate or any of the other dozens of apps that help monitor food intake. Similarly, if she wants to use the Fitbit Community social features, all of her teammates with whom she wishes to compare data must have a Fitbit wearable device, or the Fitbit app at the very least. Table 2 shows some of the common data types that are tracked by different fitness and nutrition apps. With so many different apps to track the same types of data, it is unlikely that all members of the basketball team use the same application, which limits group comparison.

	Step count	Weight	Calorie intake	Calories burned	Heart rate	Time active	Sleep
Apple Health	X	X	X	X	X	X	X
Calorie Counter - MyFitnessPal	X	X	X				
Fitbit	X	X	X	X	X	X	X
Garmin Connect	X			X	X	X	X
Google Fit	X			X		X	
Health Mate - Total Health Tracking (Withings)	X	X			X		X
iHealth MyVitals		X					
MapMyRun/ MapMyWalk/ MapMyFitness				X	X	X	
Misfit	X			X		X	X
RunKeeper				X	X	X	
Samsung Health	X	X	X	X	X	X	X

Table 2: A Sample of Applications and the Types of Data They Track

Adapted from (Apple Inc., 2019; Under Armour, Inc., 2018; Fitbit, Inc., 2018; Garmin, 2019; Google, *n.d.*, Withings, 2019; iHealth Labs, Inc, 2018; Misfit, *n.d.*; ASICS Digital, Inc., 2018; Samsung Electronics Co., Ltd., 2018)

If members of a friend group have different devices, they may only be able to track their health and fitness progress independently, which has been shown to be less effective than tracking progress and comparing with others (Consolvo, Everitt, Smith, & Landay, 2006). Through this project, we sought to make exercise and health tracking more social because we believe it will yield better results for the Worcester Polytechnic Institute (WPI) community than monitoring health independently.



## **1.5 The Goal of This Major Qualifying Project (MQP)**

We set out to eliminate the fragmentation of health and fitness data for the students and faculty of WPI, and to help the community become healthier together, regardless of the apps or devices people use. The mission of this project was to create a healthier WPI community by facilitating community members' access to health-tracking tools designed to improve their fitness. By eliminating barriers created by separate applications with isolated data stored in different formats, we created an accessible support network to encourage students and faculty to set and meet their health and fitness goals.

The goal of this MQP was to create a proof of concept for an application that will allow WPI community members to store all their health-related data in one place and compare it to that of their friends and the community as a whole, in order to help improve their overall health. To achieve our goal, we came up with the following three objectives:

1. Conduct user surveys to determine which health and fitness applications and devices the members of the WPI community currently use to stay in shape.
2. Build an Android application that compiles data from as many of the popular applications and devices the WPI community reported using as possible, allowing them to view all of their health and fitness data in one place.
3. Aggregate fitness data from the WPI community into graphs to allow users of the application to compare their own health data to that of others.

## **1.6 Project Vision**

Since people might have different types of health data scattered across different applications and databases, it could be difficult to recognize correlations between daily habits

and overall health. Certain health related risks could be easily detected with gathered data. Compiling data on sleeping, eating, and exercise habits in one application could help users better monitor their progress and identify unhealthy habits. Moreover, application features based on analyzed data like gamification could help elongate the usage of an application. Figure 2 shows how this project could centralize health information into one place for analysis and easy access.

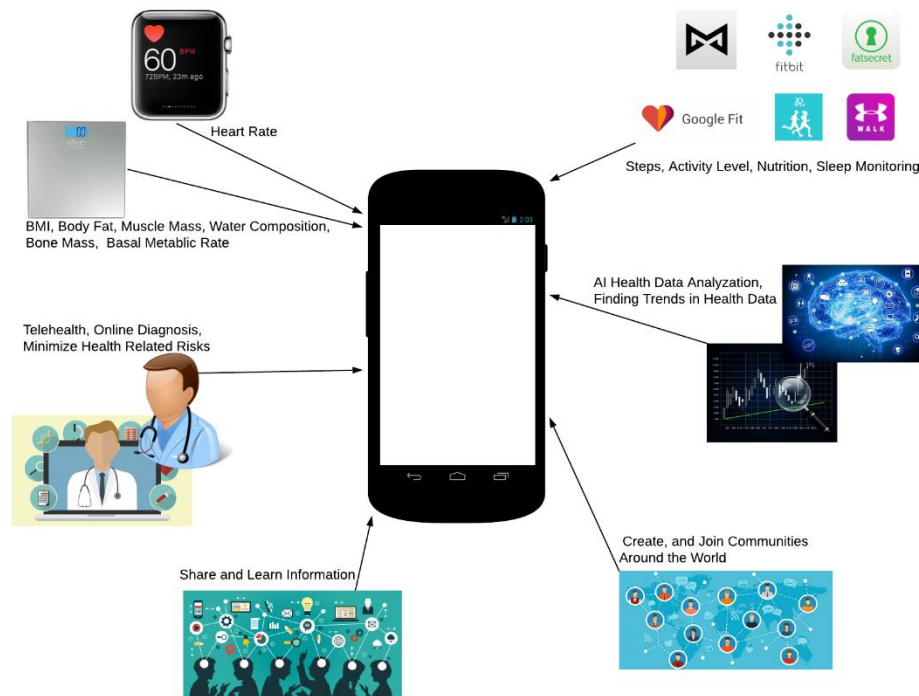


Figure 2: Illustrated Vision of HerdHealth

Adapted from (Misfit, *n.d.*; Fitbit, Inc., 2018; FatSecret, 2019; Google, 2019; Under Armour, Inc., 2019; ASICS Digital, Inc., 2018; Tofel, 2015; Khoo, 2015; Schaffhauser, 2017; Mathews, 2017; Clipart Panda, 2017; nicescience, 2017; Vive Health, 2019; Ghafourifar & Evans, 2018)

**Minimize Health Related Risks:** Compiling nutrition data could help people recognize unhealthy trends in their eating habits as well as prevent them from developing health problems,

such as micronutrient (vitamin and mineral) deficiencies. According to Bill and Melinda Gates Foundation, one third of the world's global population is affected by micronutrient deficiencies, while more than two million people die from high cholesterol each year (Bill and Melinda Gates Foundation, 2011). A method for tracking the nutritional value of food consumed might help users gain valuable feedback about their current diet, and could easily be achieved through the use of smartphones. Users would be able to upload data and obtain immediate feedback, including suggestions for improvement.

**Find Trends in Daily Habits:** In 2012, the Mobile Health Mashups group developed an Android application that collects and analyzes health and fitness data in order to find correlations that were hard to spot by simply looking at graphs and charts (Tollmar, Bentley, & Viedma, 2012). With recent advancements in artificial intelligence and machine learning algorithms lies the potential for expansion upon this work. The insight gained through a compilation of fitness and nutrition data has the potential to help users find trends in their daily habits, which will hopefully enable them to find areas of improvement to become healthier and more active.

**Increase User Engagement:** With the advancement of smartphones and technology, there are a lot of low cost devices and apps that make gathering and tracking health data, such as daily step counts, sleep quality, and heart rate easy. Several studies have found that with just wearable technology without any sort of behavioral reinforcement, users did not sustain their engagement for much longer than 6 months (Ridgers, McNarry, & Mackintosh, 2016; Finkelstein et al., 2016.) However, with the addition of the positive feedback and gamification features provided by smartphone apps, wearable device users are more likely to stay engaged over a longer period of time (S.Asimakopoulos, G.Asimakopoulos, & Spillers, 2017).

**Our Plan:** For our project, we created an application where users could gather all of their health and fitness data in one place, and implementing some of Behavior Change Techniques (BCTs) to encourage healthy habits. Charles Abraham and Susan Michie came up with 26 ways of modifying behavioral habits based on theoretical frameworks, which they termed Behavior Change Techniques (Abraham & Michie, 2008). With a set of generally applicable definitions, it became easier to analyze the effectiveness of interventions and replicate them for further study. Initially, we used five of the BCTs introduced by Abraham and Michie, but we would like to integrate more in the future. The five BCTs we will be using can be found in Table 3.

<b>BCT</b>	<b>Description</b>	<b>Our Application</b>
Prompt Intention Formation	Encourage users to set and act on a general goal	Suggest that users make daily and weekly exercise and nutrition goals
Provide General Encouragement	Praise users' performances without having specific performance standards	Give compliments when users regularly check their statistics and input data
Set Graded Tasks	Set easy tasks and increase the difficulties once the goals for the set tasks are met	Suggests that users set smaller goals (e.g. run for one hour in a week, exercise 10 mins per day) and gradually increase the goals once user meets them
Provide Contingent Rewards	Praise and reward users for specific behaviors and achievements	Praise users for specific exercise activities or for exercising more than usual
Provide Opportunities for Social Comparison	Monitor others in one's community or group	Display statistics of user's friends and where the user stands on a community leaderboard

Table 3: Details of BCTs We Plan to Use (Abraham & Michie, 2008)

Our concept could be applied to many different communities, but we decided to start at WPI. In the 2016 edition of Princeton Review, WPI was ranked 11th among U.S. colleges in terms of time students spend studying (The Princeton Review, 2015). Since WPI students spend so much time studying and working on various projects, it is possible that they do not spend as much time exercising as students at other universities. For this reason, we felt that WPI could benefit from an application like ours.

## **2. Literature Review**

In this chapter, we will discuss background information on various aspects of our project, including studies on the effectiveness of technology-based interventions, the effect of social comparison and competition on motivation to exercise, community features of fitness applications, and some background on a similar project.

### **2.1 Studies on Effectiveness of Technology-Based Interventions**

Several studies have suggested that fitness devices and applications have the potential to effectively encourage users to exercise and improve their overall wellbeing, which we aimed to accomplish with HerdHealth. Litman et al. (2015) conducted a survey of 726 participants, including some who had never used an exercise application, some who had used an application but had since stopped, and some who were currently using an application, found that current users were more physically active during leisure time than both past users and non-users. Metabolic equivalent of task, or MET, is a unit that describes the amount of energy expended during physical tasks. An estimation of MET values across the seven days prior to the survey showed that current users were nearly twice as active in their leisure time as past users, and more than twice as active as non-users. Current users also had a lower body mass index (ratio of height to weight), or BMI, than past users and non-users, but this was attributed to the fact that they exercised more (Litman et al., 2015).

A review of 27 different studies which used a smartphone application to improve diet, physical activity, and/or sedentary behavior found that 19 of the studies showed overall improvement of health outcomes. 12 of the 27 studies reported significantly greater healthy behavior improvement in the app intervention group than in the control group. Overall, the review suggests a modest potential for fitness applications to increase motivation for healthy

behaviors, though more research is needed to determine the best combination of application features to provoke healthy lifestyle changes in users (Schoeppe et al., 2016).

A pilot study involving 128 participants compared the effectiveness of a weight management intervention delivered on a smartphone app to that of website- and paper diary-based interventions. Participants were divided into three groups of approximately equal size, and each group received one of the three interventions to use over the trial period. The study found that retention rates after six months of use were significantly higher in the smartphone group than in the website or paper diary groups; 93% of participants in the smartphone group returned for a follow-up session after six months, while 53% returned in the diary group and 55% returned in the website group. Each group's overall adherence to its corresponding intervention is shown in Figure 3, where "MMM" stands for MyMealMate, the name of the smartphone application. Additionally, those in the smartphone app group lost an average of 4.6 kg over the six months, while those in the diary and website groups lost an average of 2.9 kg and 1.3 kg respectively (Carter et al., 2013).

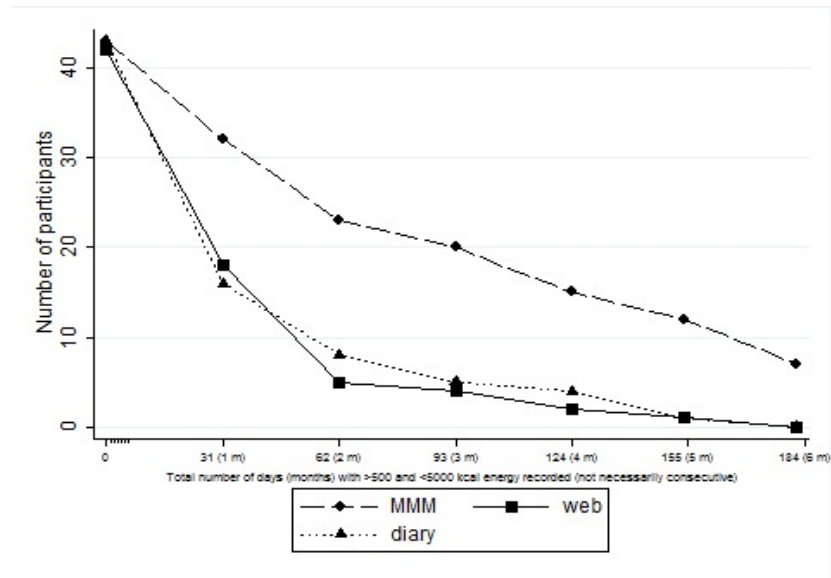


Figure 3: Number of Days Intervention was Used by Participants in Each Group  
(Carter et al., 2013)

Miragall et al. (2017) conducted a study on the effect of an Internet-based motivational intervention (IMI) and a pedometer on physical activity habits of insufficiently active participants, ages 18 to 40 years old. The researchers randomly split the participants up into three groups: the control group, one group given a 45-minute IMI, and one given the IMI in addition to access to data from a Fitbit One pedometer. All participants were given a pedometer, but only those in one group could access the data collected by the device. This group also had training on how to use the Fitbit website to find activity data and set goals. The study lasted for three weeks, and there was one additional follow up week three months after the initial study to see if the participants had changed their exercise habits in the long term (Miragall, Domínguez-Rodríguez, Navarro, Cebolla, & Baños, 2017).

The results of this study showed significant increases in daily step totals for the groups who went through IMIs. Members of the group given access to the Fitbit data improved their



step counts by an average of 35% (2,069 steps) at the end of the three week study, and an average of 38% (2,227 steps) at the follow up. The group given the IMI without the Fitbit increased its average daily step count by 18% (1,050 steps) post-intervention, and 26% (1,493 steps) at the follow up. The control group's step count also increased, but far less drastically: 7% (402 steps) after the study and 2% (94 steps) at the follow up (Miragall et al., 2017).

## **2.2 Impact of Social Comparison and Competition**

Research has shown that simply carrying a pedometer or wearable device every day is usually not enough to keep people motivated for a prolonged period of time (Speck & Harrell, 2003); additional self-influence strategies are often necessary for people to prevail over their old habits and show significant improvement (Bandura, 1986). Self-efficacy, the belief in one's ability to plan and achieve goals, is a key factor in motivation and action. One of the major ways of building self-efficacy is by watching other people that are similar to oneself succeed (Bandura, 1986). The ability to share and compare individuals' progress may help make users' fitness lives more satisfying and motivating (Consolvo, Everitt, Smith, & Landay, 2006).

An approach to addressing the problem of lacking motivation to exercise is through providing opportunities for social comparison, which is one of the BCTs we utilized. Social comparison often plays a significant role in motivating individuals to improve their performance. Studies show that people put in more effort for a task when there is a "competitor" involved, whether that person is in direct competition or simply acting as a means for comparison (DiMenichi & Tricomi, 2015).

Another study shows how goal achievement is influenced by several other factors, one of which is the effect of sharing one's goals with others compared to writing down goals and

keeping them to oneself. Over 70% of the participants in the study who sent weekly updates to a friend reported successful goal achievement (completed their goal or were more than halfway there), compared to 35% of those who kept their goals to themselves, without writing them down (“Study focuses on strategies for achieving goals, resolutions,” *n.d.*). This suggests that a social platform for comparing and sharing goals might keep people accountable and help them achieve their fitness goals. Finally, another study supports this finding by showing that social support and goal-setting increase physical activity, which is one of the main purposes of fitness apps and wearable devices (Jiryae, Siadat, Zamani, & Taleban, 2015). In response to all this research, we planned to implement social features into HerdHealth to increase motivation to exercise among WPI community members.

### **2.3 Community Features of Fitness Apps**

Many fitness apps already implement the “provide opportunities for social comparison” BCT through community features that allow users to interact with each other. Fitbit, a popular app for tracking step count and other statistics, allows users to share daily step counts and other personal stats with the community, as well as post to a forum to get advice and responses from other users. Users can also join groups to share and compare their stats with a specific group of friends or contacts (Kosecki, 2017). Similarly, MyFitnessPal, an app for tracking calorie count and nutrition, has a social feature that also includes options to create groups and forums to ask questions and meet others in the community. In addition, MyFitnessPal has a fitness challenges category, where users can join discussions based on certain challenges they are interested in. For example, if a user is interested in planking to improve core strength, he or she can join a forum for a challenge to complete 60 minutes of planks in a month. Users who join this forum can post

about their own progress with the challenge, as well as encourage each other to complete their daily planks. This is a particularly effective way to motivate individuals, since they are held accountable by others to reach their goals and improve their fitness (“Welcome to the MyFitnessPal community!,” 2017). These social features served as inspiration for the community implementation in HerdHealth.

## **2.4 Open mHealth**

Open mHealth is a free online service that provides open source tools for integrating data from various health and fitness devices and apps into a standardized format. The tools include a set of schemas that define a consistent structure for common types of health care, as well as an Application Programming Interface (API) that allows for centralized storage of health and fitness data, along with restriction of access for heightened security (Open mHealth, 2015b). Another one of the tools is Shimmer, an application that facilitates pulling data from third party APIs and converting it to a format compliant with Open mHealth’s schemas (Open mHealth, 2018). While we did not use Open mHealth’s tools directly in HerdHealth, the ideas behind Open mHealth served as a basis for the concept of this project.

### *2.4.1 Open mHealth Vision*

The vision behind Open mHealth is a world where people can easily link health data from separate devices and share it with their doctors. Digital data would be accessible, for free, to anyone who needed it, and it could be stored in one place for ease of transfer (Open mHealth, 2015a).

The concept first appeared in a policy paper written by Deborah Estrin and Ida Sim (Open mHealth, 2015a). The authors noted that mobile health applications have the potential to determine optimal treatment for chronic diseases as quickly as possible, but this potential is limited if each existing health application is closed, with its own data format that other programs cannot share. Rather than this “siloesd” approach, they suggested that an open architecture, one accessible to anyone interested in using or expanding its functionality, could lead to a broad community participating in app design. Additionally, with more anonymous standardized health data from consenting users available in one place, health research and clinical care could experience significant advancements; they cited an example in which every patient prescribed an antidepressant is invited to participate in a study. Data such as activity levels and records of side effects could be collected from the participants and used to determine the effectiveness of the antidepressants. The Internet’s architecture, which is not only open but includes a “common Internet protocol for transferring data,” sparked its massive success and groundbreaking applications, and so, the authors argued, mobile health could benefit from a similar structure (Estrin & Sim, 2010). Figure 4 below contrasts the “siloesd” means of data management in health applications with Estrin and Sim’s concept of an open architecture. Open mHealth’s architecture would funnel data into a common storage format and API, as opposed to the separated data and multiple APIs in the “stovepipe” approach. These benefits of the open architecture allow for much simpler development of applications that require data from multiple sources.



Figure 4: Diagram of Structure of Siloed (Stovepipe) Approach vs. Open mHealth

Estrin and Sim, along with five other researchers, elaborated on the vision behind Open mHealth in an article in the *Journal of Medical Internet Research* from 2012. They stated that in order to accomplish its goal of allowing mobile health to reach its full potential, Open mHealth must have five features. Although we did not use Open mHealth in our project directly, we adapted two of these features for use in our application: “Flexible architecture: recognizes both the limits and the utility of existing closed systems and is designed to maximize participation from all players,” and “Iteration: delivers efficient reuse through collaborative cycles of development” (Chen et al., 2012). A flexible architecture similar to that of Open mHealth

allowed us to integrate APIs from multiple applications into a single set of standardized data, and an “iteration” property meant future MQP teams could build on our application. Additionally, while we did not plan to use the data we collected for clinical research, we believed the data could still provide useful findings. For example, knowing the average fitness level of the WPI community could lead to the possibility of fitness programs tailored to students’ and faculty and staff members’ needs.

## **2.5 Our approach**

Our approach to motivating people in the WPI community to live healthier lives through the use of a mobile application was to apply similar concepts that other platforms and devices currently use, such as tracking exercise and calories burned, and pool together data from multiple sources. That way, users would have a centralized place to view all of their statistics from their apps and devices.

HerdHealth includes a home screen to display a user’s data, a statistics page where users can compare their data to that of the WPI community as a whole, a page where users can set their goals, and a leaderboard that shows which users are doing the most to improve their overall health. The data types we display are steps taken, calories burned, distance traveled, calories consumed, sodium intake, fat intake, and protein intake. The app can pull data from Fitbit, Misfit, MapMyWalk, FatSecret, and RunKeeper. A visualization of which APIs outline to which data types appears in Figure 5. In upcoming chapters, we will explain how we used our approach to build our platform.

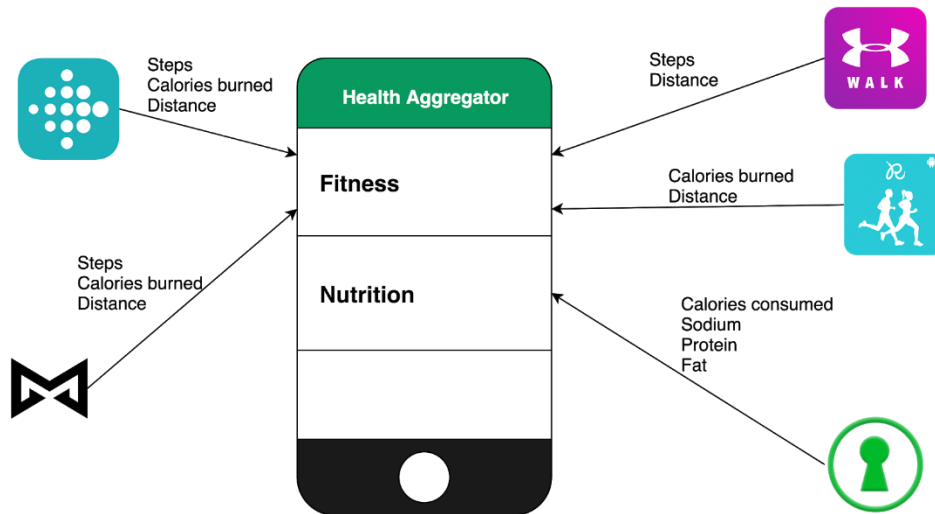


Figure 5: Diagram of Fitness APIs and Data They Provide

In addition, plans for our HerdHealth application included a social feature, where users would be able to utilize the “provide opportunities for social comparison” BCT regardless of the different apps and devices they use to track their data. The idea was that users could also share data within specific groups, including students in their graduating class, major, or clubs, to motivate each other to continue improving. However, due to time constraints, we were unable to implement this social feature.

### 3. Survey

In order to determine what data to include in HerdHealth and which fitness apps and devices to pull it from, we needed to know how the WPI community used fitness apps. To gain this insight, we created a survey. A flow chart representing the survey's questions and logic can be found in Appendix 1. The survey asked which fitness apps and devices people used, how often they used them, and how interested they would be in the application we planned to write. Our goal was to gauge interest in the application overall, as well as understand what specific features would benefit the WPI community most. Namely, we wanted to find out which applications and devices to support in creating HerdHealth, what types of fitness data (step count, calorie count, etc.) to include, whether or not students and faculty were interested in social features to engage with each other, and if there were any additional features members of the community wanted to see in our mobile application.

We started by asking about the fitness apps and devices that people used. In the first question, we asked about the frequency with which people used the following apps: Fitbit, Google Fit, RunKeeper, Misfit, Withings, iHealth, and MyFitnessPal. We chose to ask about these apps because their APIs (with the exception of MyFitnessPal's) were supported by Shimmer, the application we originally planned to use to help us pull data from each of these sources. We included MyFitnessPal even though it was not supported by Shimmer because we knew that it was a popular application for tracking dietary and exercise habits. We asked respondents to report how often they used each of the applications and devices, using one of the following indicators: "never," "once per month or less," "several times per month," "once per day," or "several times per day/constantly."

Next, we asked if the participant used any health-oriented apps or devices not mentioned in the previous question, and if so, which ones. Afterwards, respondents were asked to rate how



often they used each additional app just as they had in the first question. From the responses to this question, we hoped to gain an understanding of which apps and devices not supported by Shimmer would be most beneficial to support in our application.

The next questions were intended to see if respondents used social features in any fitness apps or devices they had previously reported using. From these questions, we hoped to gather information about the usage of social features of existing fitness apps to gauge interest in a social feature in the application we planned to develop.

We then asked about what the participants were looking to get out of using their fitness apps and devices. In particular, we wanted to know which specific health indicators they used apps and devices to track. We asked this question in order to get an idea of what kind of data would be available to us if we gathered data from the apps and devices that members of the WPI community used.

Next, we asked how interested respondents were in improving their health and fitness. We wanted to know the particular areas, if any, in which people wanted to improve. In doing so, we could get an idea of what data people would be most interested in viewing in order to improve their own health.

The following question gauged overall interest in HerdHealth. We wanted to know if people would use a mobile app that would allow them to view and track their own health and fitness data, as well as compare their statistics to those of other members of the WPI community, should they choose to do so. The question made it clear that the data would be private only to the user and other approved users. This was so the respondent was not worried about private data being open for everyone in the community to see. We also asked about the specific types of data

users would like to compare to that of their friends, so that we could gain insight into what types of data we should include in the social features of our application.

Finally, we asked respondents if there were any additional features they would like to see in our app, in order to find out what other features we could include that might be useful for members of the WPI community who were looking to improve their health and fitness.

The remaining questions in our survey were demographic-related. We hoped to gather some information about the variety of survey respondents, to see if it was representative enough of the WPI population as a whole. Also, this could help us determine if there were any patterns among users of fitness apps and devices, and who specifically to keep in mind when designing our mobile app.

### **3.1 Survey Results**

This section will outline the results of the survey we administered prior to beginning the development of HerdHealth.

#### *3.1.1 Demographics*

Out of the 251 respondents who answered the question regarding their status on campus, 45.8% (115) were WPI students, 17.5% (44) were WPI faculty members, and 36.7% (92) chose “other.” The vast majority of those in the “other” category wrote that they were staff. After reorganizing the results, we determined the breakdown to be that shown in Figure 6.

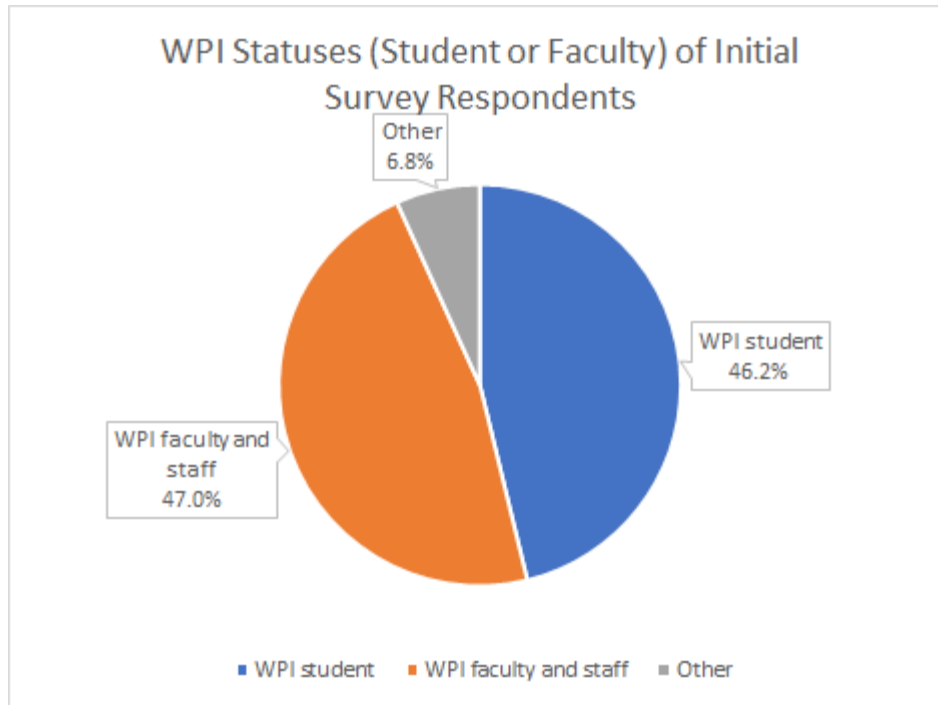


Figure 6: Statuses (Student, Faculty/Staff, or Other) of Initial Survey Respondents, N=251

Figure 7 shows the percentage of respondents who stated that they were male, female, “other”, or “prefer not to say.” Out of 251, 36.7% (92) were male, 61.0% (153) were female, 1.6% (4) chose “prefer not to say”, and 0.8% (2) chose “other.”

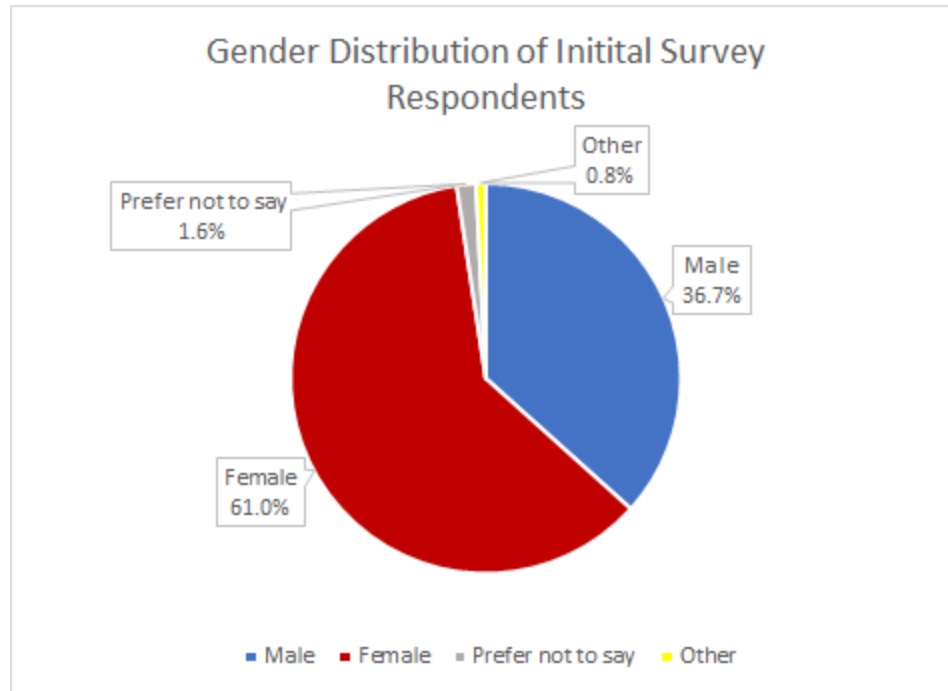


Figure 7: Gender Distribution of Initial Respondents, N=251

Figure 8 shows the age distribution of respondents, separated by age range and including a “prefer not to say” category. The most commonly reported ages fell into the 17-22 range; they accounted for 44.0% (110) of the respondents. This was likely because most WPI students fall into this age range, while faculty members’ ages are distributed over a much wider range of ages.

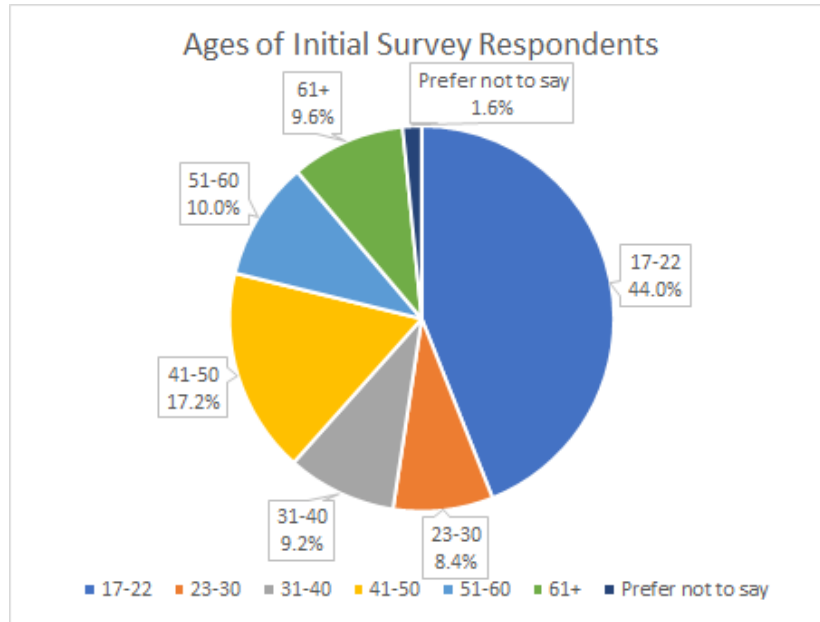


Figure 8: Ages of Initial Survey Respondents, N=250

Figure 9 shows the self-reported activity level of respondents. Of 251 total, 18.3% (46) chose the “very active” response, 34.7% (87) chose “active”, 43.0% (108) chose “lightly active”, and 4.0% (10) chose “not active.”

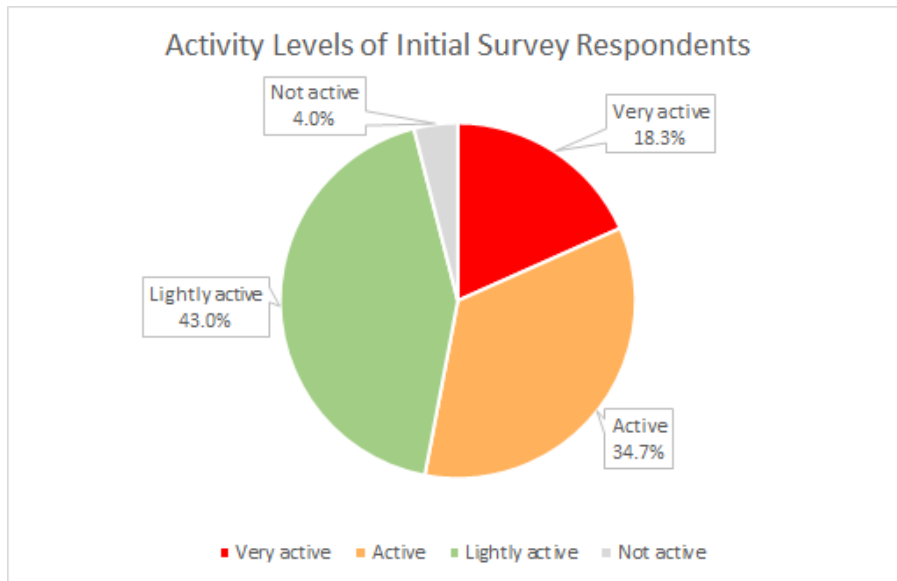


Figure 9: Activity Levels of Initial Respondents, N=251

Figure 10 shows the types of mobile phones owned by the survey respondents. Most owned either an iPhone or an Android; out of 251 respondents, 66.9% (168) owned an iPhone, 31.5% (79) owned an Android, and 1.6% (4) did not own a smartphone.

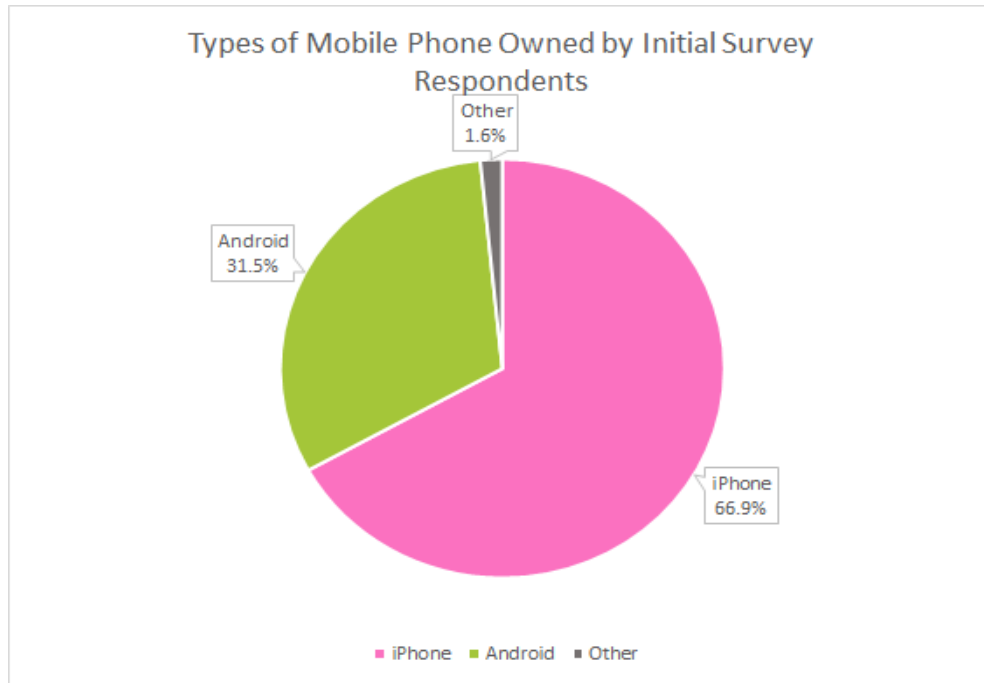


Figure 10: Types of Mobile Phone Owned by Initial Respondents, N=251

### 3.1.2 Apps and Devices Used by Respondents

We received 261 responses from both WPI faculty and students and compiled the responses to the questions about how often people used specific fitness apps. Figure 11 shows how many of the respondents reported using specific fitness apps more often than “never.” The results include both the applications that we provided in the first question and the four most commonly entered additional applications from the second question. Some respondents manually entered applications that we included in the list. Originally, we planned to integrate data from all the applications shown in the figure. However, once we decided to use GetHealth, we were limited to only the APIs available through the service. These included several of the applications

listed in the graph; in particular, HerdHealth’s supported applications include Fitbit, GoogleFit, MapMyWalk, Runkeeper, and Misfit.

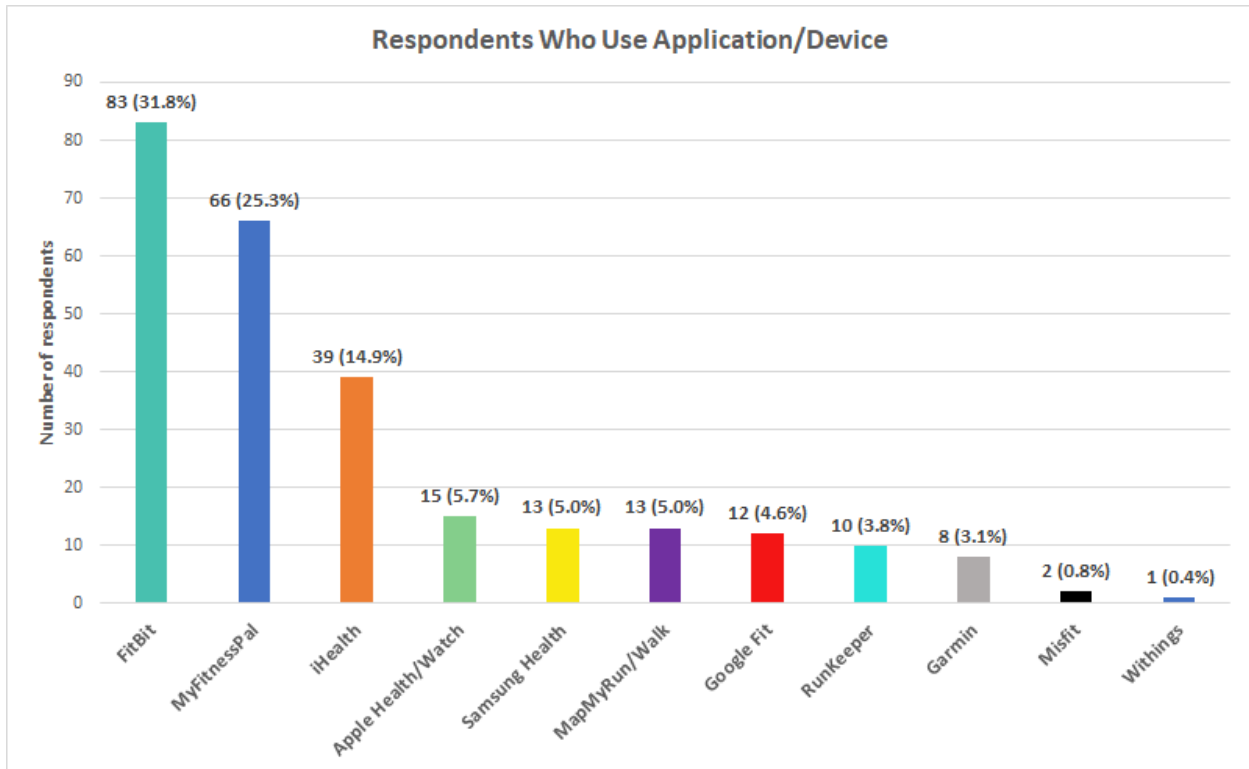


Figure 11: Respondents Who Use Application/Device, N=261

### 3.1.3 Social Features

Out of the 261 respondents, 80.8% (211) answered the question “Do you use any social features built into your fitness/health apps, if any exist?” Of those, 14.7% (31) answered “Yes,” and 85.3% (180) answered “No.” We then asked the people who reported using the social features if they thought it helped them improve their health or fitness, to which 65.4% (17) said “Yes.” The responses to these two questions are shown in Figures 12 and 13, respectively. Responses given for how the social features have helped included “motivation to keep up with friends,” “more accountability,” and “encouragement.” While the majority of those who

answered the question did not use their applications' social features, more than half of those who did reported that they were helpful. Therefore, we decided a social feature would be a worthwhile inclusion in HerdHealth.

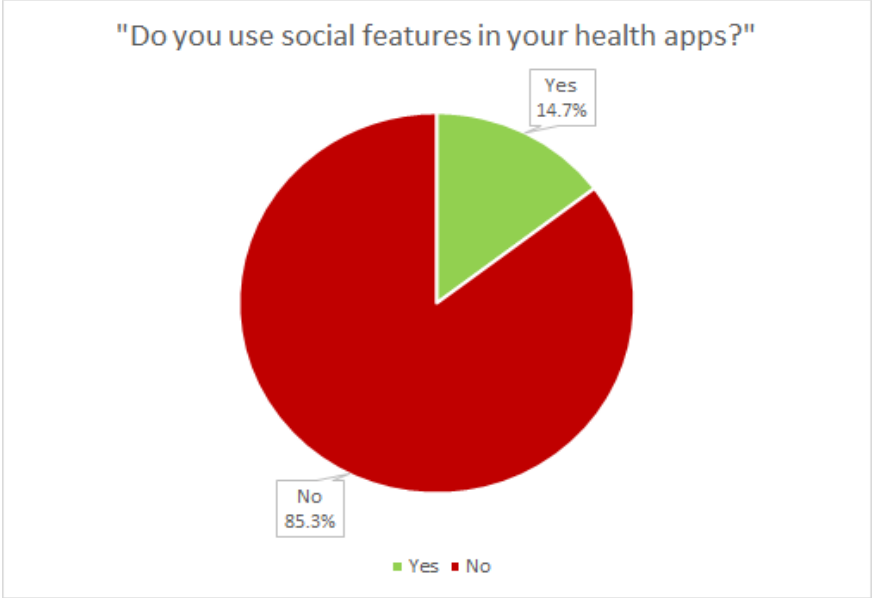


Figure 12: Do You Use Social Features in Your Health Apps?

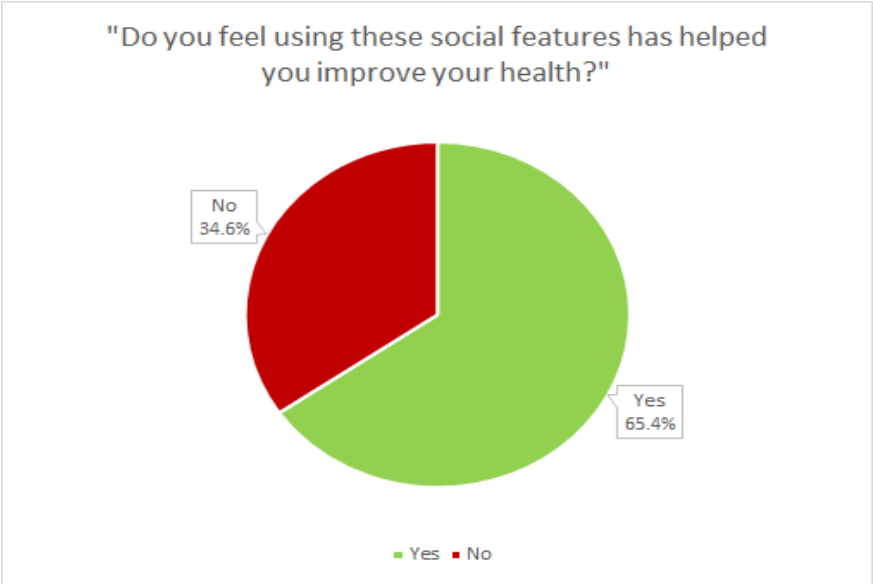


Figure 13: Do You Feel Using These Social Features Has Helped You Improve Your Health?



### 3.1.4 Types of Data Tracked

Figure 14 shows how many of the respondents tracked each type of data. Note that a respondent could choose more than one. The most commonly tracked health indicator was step count, followed by “Aerobic activity (i.e. walking, jogging, biking, etc.)” It is important to note that the latter category indicates a wide range of specific indicators, including distance, speed, and heart rate; this is likely why it was one of the most common responses.

Twenty-five respondents chose “other,” but of those, ten either wrote “none,” “N/A”, or “I don’t use these devices” when prompted to enter what they use the devices to track. Other responses included distance, heart rate, “measurements,” swimming, active minutes, number of days per week with at least one exercise, steps per hour, “wellness/mental health/self care,” randomized yoga practices, stress, and stairs. We wanted to prioritize supporting the data types with the most responses, such as step count and calories burned, if possible, while other data types could be options for future work.

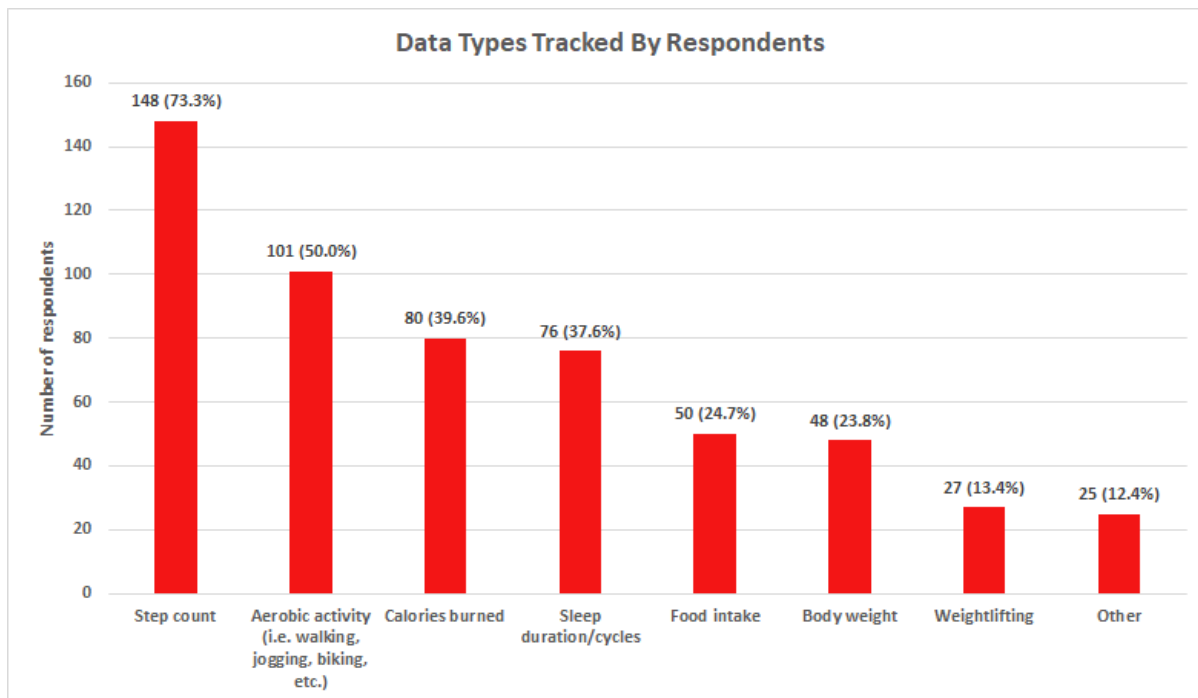


Figure 14: Data Types Tracked by Respondents, N=202

Figure 15 shows how many of the respondents expressed a desire to improve their health or fitness in each of the given areas. As in the previous question, a respondent could choose more than one. The area with the most responses was “Aerobic activity (i.e. walking, jogging, biking, etc.),” closely followed by body weight. Only seven respondents chose “other.” The text entries of those who did choose “other” were “Heart rate monitoring,” “Strength Training,” “BMI (Body Mass Index),” “blood pressure,” “Strength & Endurance,” “General movement during day,” and “body composition.” Similarly to the question asking which data types respondents tracked, the responses to this question indicated which data types would be best to include in HerdHealth. Again, it is likely that “Aerobic activity” was a common response because it encompasses many different types of data.

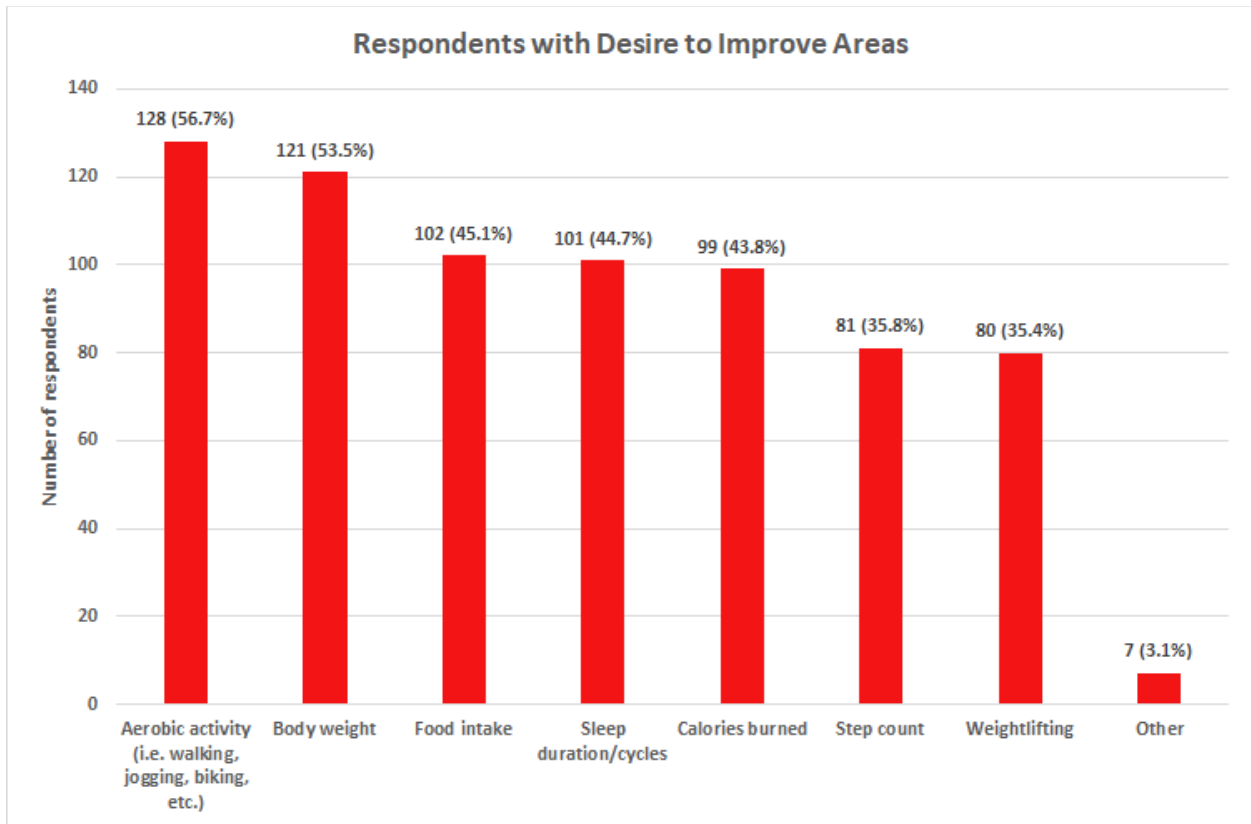


Figure 15: Respondents with Desire to Improve Areas, N=226

Figure 16 shows how many respondents said they would like to compare each type of data with their friends using our application specifically. The distribution of responses resembled that of the question that asked which areas they already used their apps and devices to track. Because of this similarity, we determined that all data a user could view in HerdHealth should also be available to compare with friends, should the user choose.

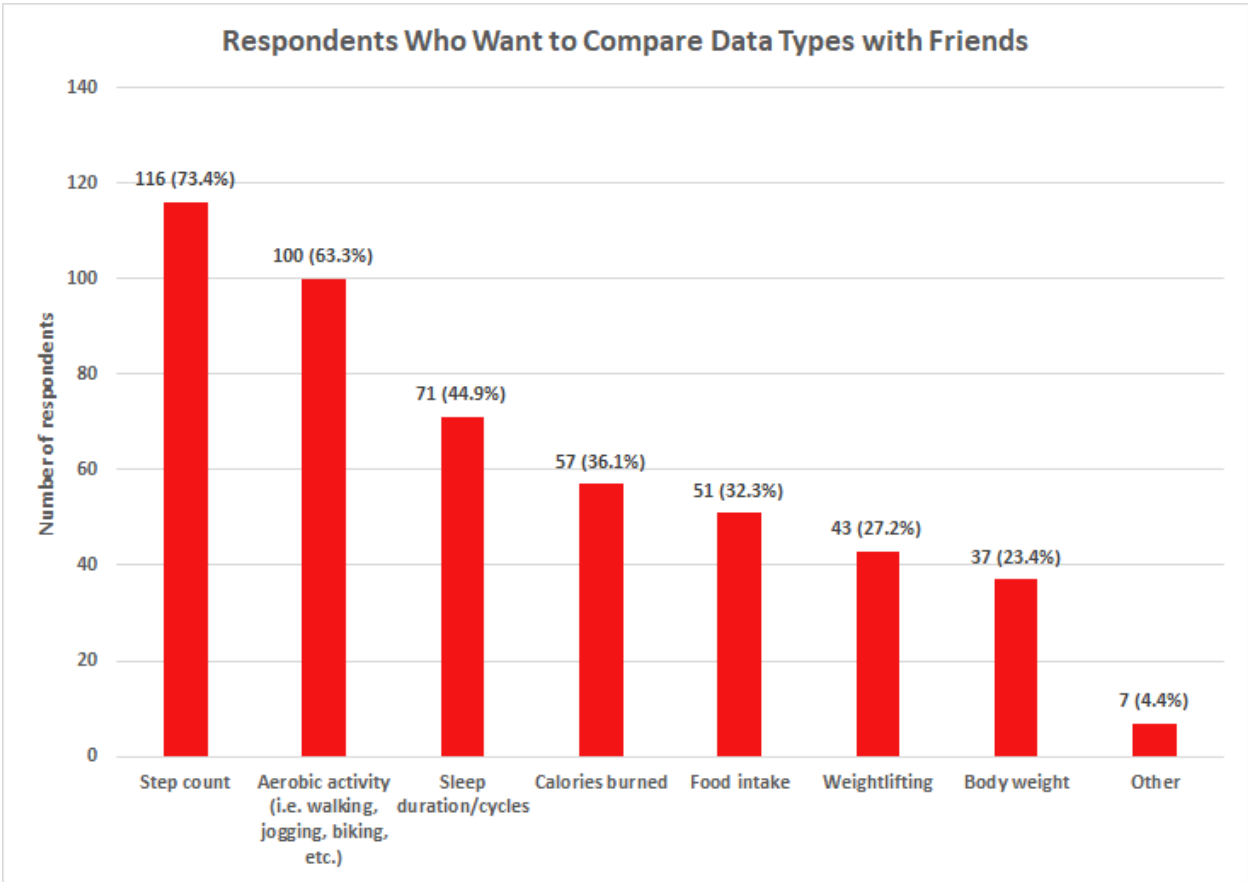


Figure 16: Respondents Who Want to Compare Data Types with Friends, N=158

3.1.5 Interest in HerdHealth

Of the 257 respondents who answered the question about their interest in our application, 73.2% (188) reported at least a slight interest; that is, they chose an answer other than “not interested at all.” While it is possible that some respondents were able to guess the intention of

the survey and only expressed interest to be polite, we believed these results indicated enough interest in the application to warrant moving forward with the project. The exact distribution of the responses can be found in Figure 17.

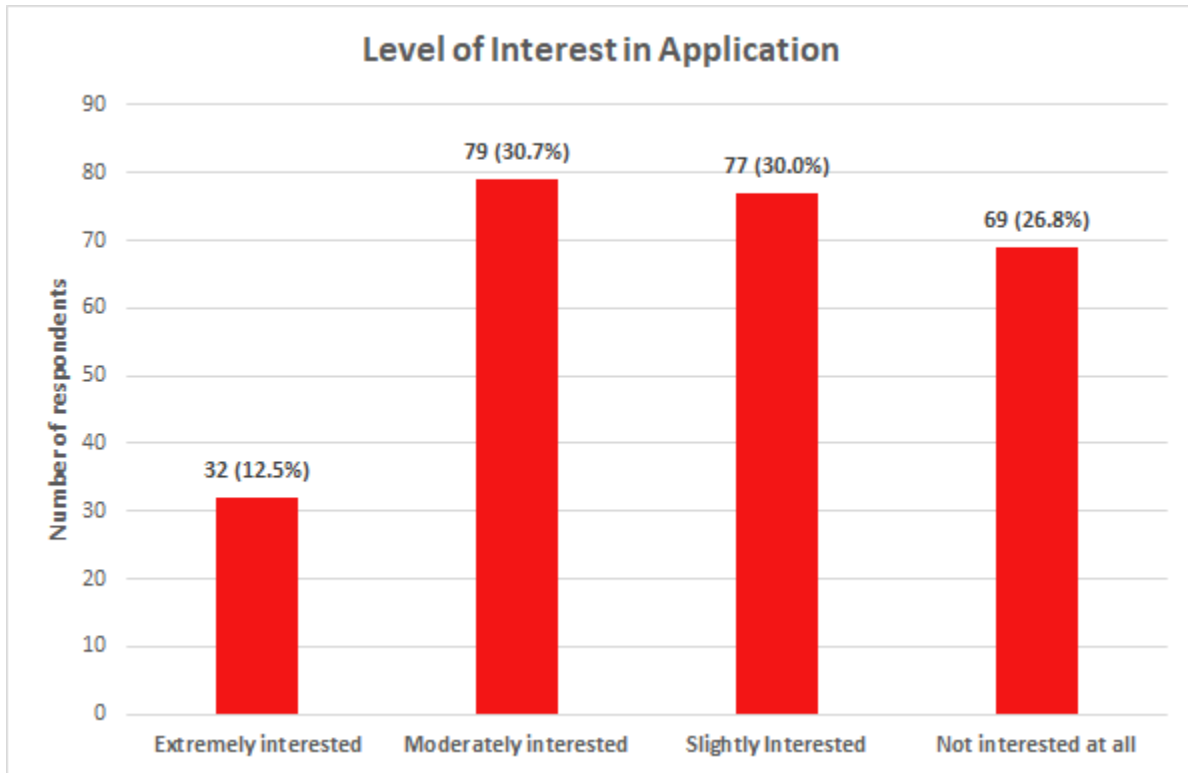


Figure 17: Level of Interest in Application, N=257

Of the 173 of those who responded to the question asking which additional features they would like to see in HerdHealth, most (86.1%, or 149) responded that there were none. While each of the 24 suggested features was different, some were similar. For example, two respondents proposed a reward system to motivate users. Two suggested the application address stress in some way, with one simply writing “stress levels” and the other proposing “guided meditation and deep breathing techniques.” While we did not implement either of these features in the HerdHealth prototype, both could be incorporated into future work.

A summary of the survey results can be found in Table 18 below.

Question	Responses, from most to least common (percentage)	Implications
<p>“How often do you use the following apps/devices?” and “Do you use any fitness/health apps or devices not listed in the previous question?”</p>	<p>Fitbit (31.8%), MyFitnessPal (25.3%), iHealth (14.9%), Apple Health/Watch (5.7%), Samsung Health (5.0%), MapMyRun/Walk (5.0%), Google Fit (4.6%), RunKeeper (3.8%), Garmin (3.1%), Misfit (0.8%), 1 (0.4%)</p>	<p>We wanted to prioritize supporting the applications with the most number of responses, if possible. Fitbit and MyFitnessPal would be relevant to the most users.</p>
<p>“Do you use any social features built into your fitness/health apps?”</p>	<p>No (85.3%), Yes (14.7%)</p>	<p>Though most users of fitness apps did not use the social features, enough did to warrant adding a social feature to HerdHealth.</p>
<p>“Do you feel that using social features in these apps has helped you improve your health or fitness?”</p>	<p>Yes (65.4%), No (34.6%)</p>	<p>The majority of those who used social features felt it had helped them improve, so including a social feature in HerdHealth could be beneficial.</p>
<p>“What do you use your device/app(s) to track?”</p>	<p>Step count (73.3%), Aerobic activity (50.0%), Calories burned (39.6%), Sleep duration/cycles (37.6%), Food intake (24.7%), Body weight (23.8%), Weightlifting (13.4%), Other (12.4%)</p>	<p>We wanted to prioritize supporting the data types with the most responses, such as step count and aerobic activity, if possible.</p>
<p>“Are you interested in improving your fitness or health in any of these areas?”</p>	<p>Aerobic activity (56.7%), Body weight (53.5%), Food intake (45.1%), Sleep duration/cycles (44.7%), Calories burned (43.8%), Step count (35.8%), Weightlifting (35.4%), Other (3.1%)</p>	<p>We wanted to prioritize supporting the data types with the most responses, such as aerobic activity and body weight, if possible.</p>
<p>“How interested would you be in using an application that compares fitness and health data across the WPI community?”</p>	<p>Moderately interested (30.7%), Slightly interested (30.0%), Not interested at all (26.8%), Extremely interested (12.5%)</p>	<p>There was enough interest in the application to warrant moving forward with the project.</p>

<p>“In the application...which types of data would you like to be able to compare to those of your friends?”</p>	<p>Step count (73.4%), Aerobic activity (63.3%), Sleep duration/cycles (44.9%), Calories burned (36.1%), Food intake (32.3%), Weightlifting (27.2%), Body weight (23.4%), Other (4.4%)</p>	<p>We wanted to prioritize supporting the data types with the most responses, such as step count and aerobic activity, if possible. Particularly, it would be beneficial to include them in the social feature.</p>
--	--	---

Figure 18: Summary of Initial Survey Results

## 4. HerdHealth System Architecture

We planned to build a fitness Android application that would allow us to easily pull health data from third-party APIs and include social features for users to compare their own health data with that of other WPI students and faculty. HerdHealth needed to be highly scalable and efficient to accommodate for the expanding WPI community. We used a cloud database to store information about the application and overall user base, in addition to a local database to store data relevant to the current user. Figure 19 below shows the system architecture for our application.

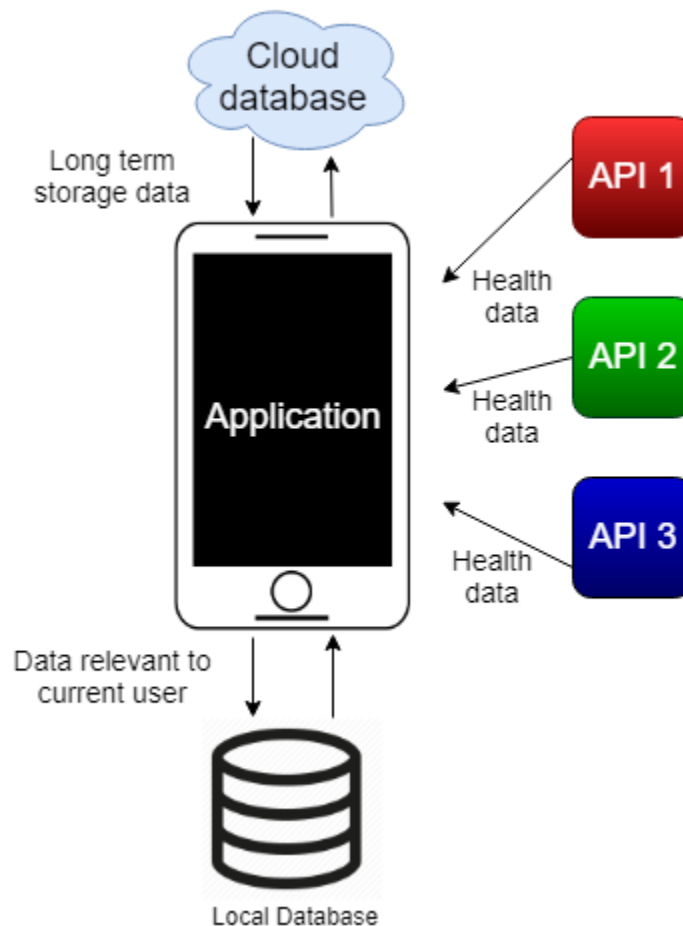


Figure 19: System Architecture Diagram

Adapted from (schoolfreeware, 2017; Database Icon, 2019)

## **4.1 Back End**

The back end of the application served to pull data from user-specified APIs, store this data in a cloud database, and store data relevant to the user in a local database. This section describes each of these functions.

### *4.1.1 Retrieving Data*

Each API had its own method of authentication, as well as its own format for storing the data. In order for the data to be effectively stored, all of the data had to be converted to a standardized format.

### *4.1.2 Cloud Database*

Information about the user base was stored in the cloud database. The application pulled data from the cloud database to retrieve information, such as community averages to display as statistics and individual users' information to display on a profile page. The code was able to read from, and write to, the database.

Data stored in the cloud database included a list of users, as well as their friends, groups of which they were a part, and their health and fitness data. For each user, the database stored some general information, such as name, email, department/major, and APIs which they connected to the app.

### *4.1.3 Local Database*

A local database was useful to store frequently-accessed information in an attempt to minimize the number of calls made to the cloud database. Information relevant to the current



user, such as the APIs to which he or she was connected and his or her own fitness data, was stored in the local database. Often, both the local and cloud databases were updated with the same information, such as when a new user was created during the signup process.

## **4.2 Front End**

When users opened the app for the first time, they were required to complete the signup process. Once the user logged in, the front end of our application was divided into four main screens: the profile screen where users could view their own data, the statistics screen where users could compare their own data to that of the community as a whole, the community screen where users were able to interact and view their friends' progress towards their goals, and the settings screen, which included a variety of functionality. Each of these screens are described in more detail in the following sections. The overall user workflow of our application is shown below in Figures 20-22. The diagrams are connected by the corresponding circled letters.

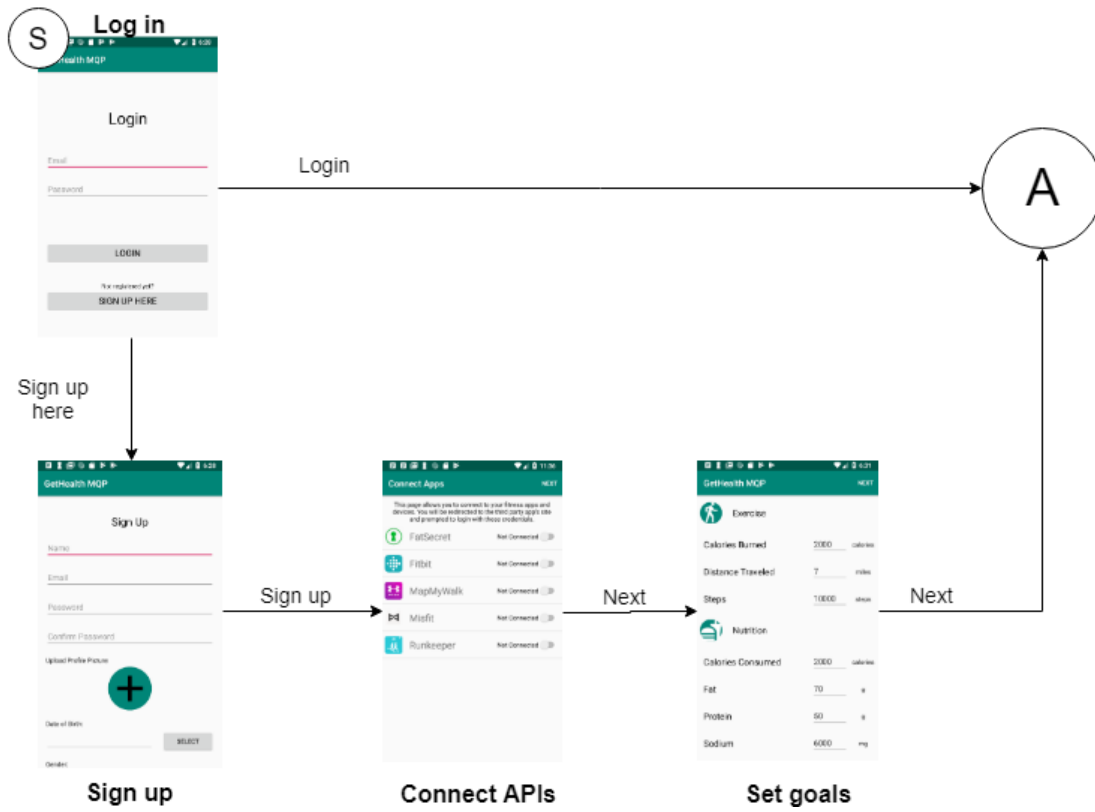


Figure 20: Workflow of Application (Part 1)

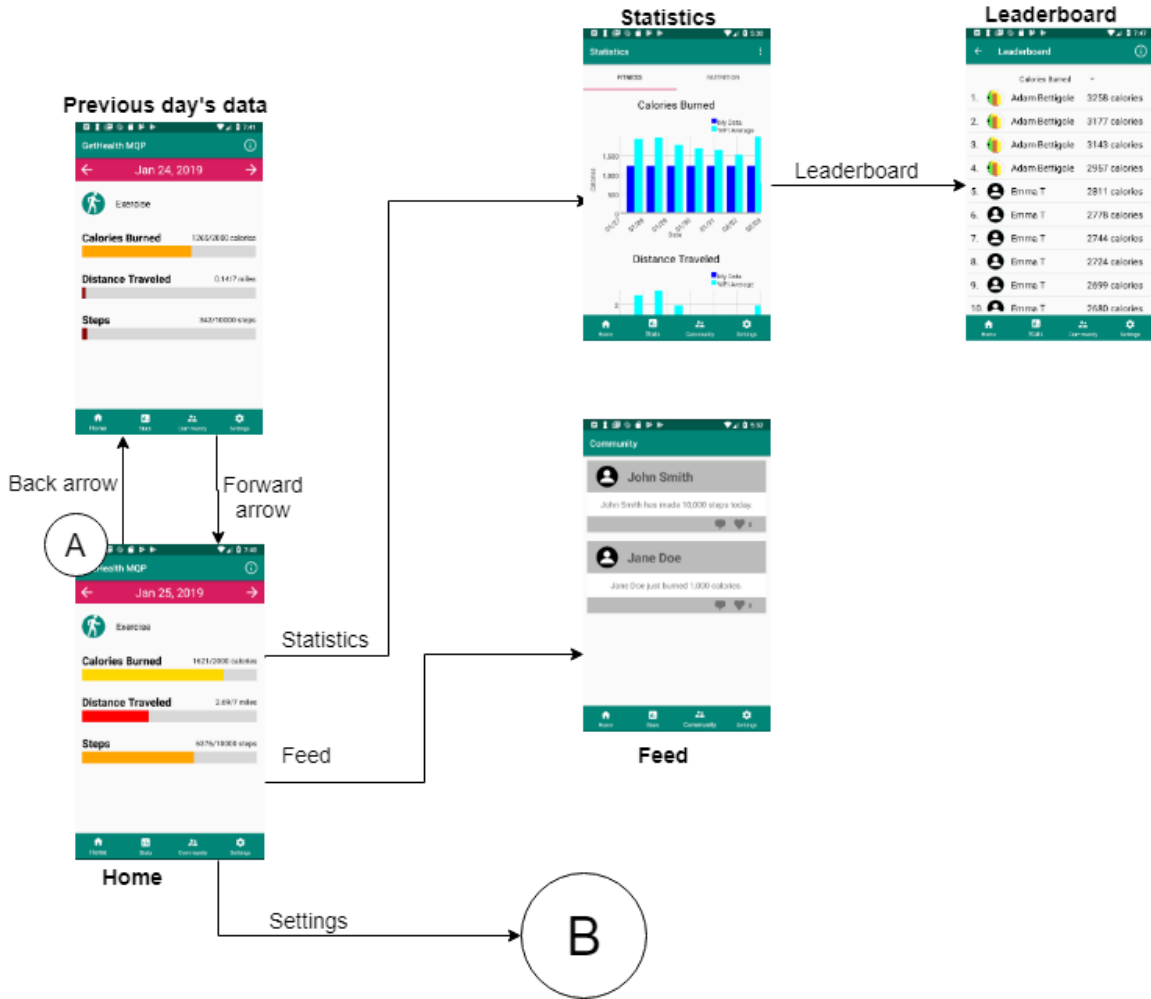


Figure 21: Workflow of Application (Part 2)

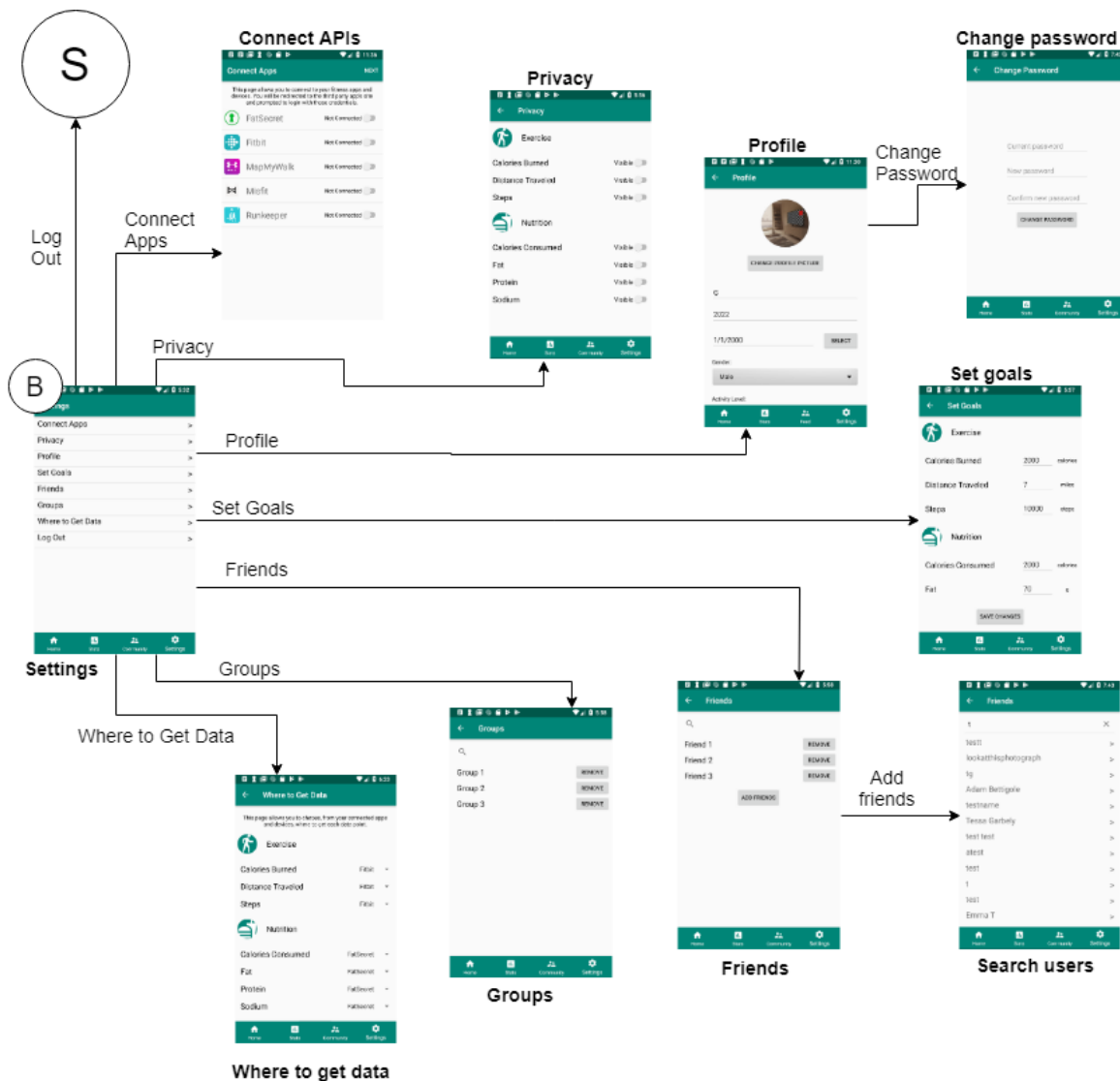


Figure 22: Workflow of Application (Part 3)

#### 4.2.1 Login and Sign Up

The first screen the user saw when opening the HerdHealth was the login screen. Users were able to log in by entering their email and password. To create an account, users had to go to the sign up screen, where they set their name, email, password, date of birth, gender, activity level, status (student or faculty), profile picture, and major (for students) or department (for faculty). After signing up, the user was given the opportunity to connect to APIs and set fitness goals.

#### *4.2.2 Home Page*

The home screen presented the user's health and fitness information, pulled from the APIs authenticated by the user. The home page also included the user's progress towards fitness goals. The layout included an expandable list that allowed the user to view their data by category (e.g. fitness data was separated from nutrition data). A bar at the top allowed the user to switch between data from different dates. Upon connecting an API, HerdHealth pulled data from the two weeks prior to the current date, so at any given time, a user would be able to view at least two weeks of their data.

#### *4.2.3 Statistics Page*

The statistics screen featured graphs of fitness data collected from the WPI community. It presented averages of step counts, calories burned, and other areas of fitness data and allowed users to compare their own data to that of the community as a whole.

The screen was divided into tabs, with each tab showing a certain category of data, such as fitness or nutrition. Each tab had a set of graphs underneath it, and each graph displayed the data of a particular type, such as step count or calorie count.

The statistics page linked to a leaderboard where users could view the highest-achieving users in fitness and nutrition categories. There were three separate leaderboards in the fitness category: one for most steps, one for most calories burned, and one for furthest distance traveled over the prior seven days. The nutrition leaderboard displayed the users who came within 10 percent of all their nutrition goals (calories consumed, sodium, etc.) for the most consecutive days.

#### 4.2.4 Feed Page

The community screen allowed the user to view friends' progress toward their health goals. Each post indicated that a user in a friend group had reached a milestone in working toward a goal. For example, one message may have read "John Doe has achieved his goal of 10,000 steps today!" Users were able to both "like" and comment on posts to encourage their friends.

#### 4.2.5 Settings Page

The settings screen allowed the user to customize various aspects of their profile, including groups they belonged to and their individual fitness goals. Users could add and remove friends on the settings page, as well as connect to APIs from which they wished to retrieve data.

The main settings screen was a list of features the user could alter. Clicking each link in the list brought the user to a separate screen focused on that particular item. In total, there were eight settings pages: *APIs*, *Privacy*, *User Info*, *Set Goals*, *Friends*, *Groups*, *Where to Get Data*, and *Log Out*. The *APIs* page, similar to the one provided in the signup process, allowed the users to connect or disconnect to the provided APIs. When connecting, the user would be taken to the API's website to log in and grant the application permission to use the data. The *Privacy* page let the user set which health data other users would be able to see. The *User Info* page provided a means for the user to change his or her information, including profile picture, name, gender, graduation year, major or department, date of birth, and activity level. *User Info* also included a link to a page where the user could change his or her password. *Set Goals* was the same as in the signup process, allowing the user to change his or her fitness goals each of the data types. On the *Friends* page, users could either add or remove current friends. They could also search all

HerdHealth users to find potential friends to add. The *Groups* page, upon completion, would have had similar functionality; users would either join or exit user-created groups or search for new groups to join. In *Where to Get Data*, the user could select an API from which to pull each data type. For example, the user may obtain fitness data from Fitbit but nutrition data from FatSecret. *Log Out* logged the user out of the application and cleared locally-stored data that was specific to the user.

## **5. Implementation**

This chapter will describe specific details of our implementation, including the third-party tools we used, our application design, development decisions made during the implementation process, and the custom Java classes and interfaces we created to arrive at our final application.

### **5.1 Android Application Design**

The backend functionality of HerdHealth is written in Java, while the user-facing front-end uses Extensible Markup Language (XML). All of our code was written in Android Studio, an Integrated Development Environment (IDE) specifically intended for programming applications for Android smartphones, tablets, TVs, and smartwatches.

### **5.2 Tools**

We used several third-party tools to help us with different aspects of our application, such as requesting, parsing, and storing data. GetHealth and OkHttp helped make requests to the different health apps and devices we supported, Gson parsed the responses from those requests, and Firebase and Realm stored all of our data. In addition, GraphView displayed our data in bar graphs and Glide assisted with displaying images. The following sections explain the capabilities of these tools in more detail and why we chose to use each one. A summary of the tools we used can be found in Table 5. Figure 23 below shows our implementation of the various libraries and APIs we used and how they're used.



Tool	Use in HerdHealth
GetHealth	Simplifies interaction with (connecting to and pulling data from) multiple APIs
Google Firebase	Authentication and cloud data storage
Realm	Local data storage
OkHttp	Makes HTTP requests to GetHealth's REST API
Gson	Parses the JSON responses retrieved from GetHealth via OkHttp
GraphView	Display graphs that compare the user's health and fitness data to the WPI community averages
Glide	Loads and displays images such as the user's profile picture

Table 4: Tools Used in HerdHealth and How They Were Applied

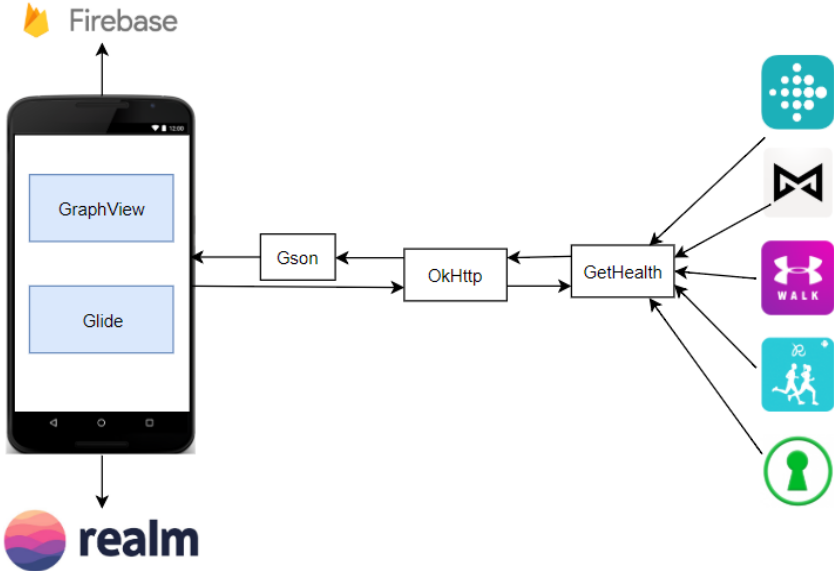


Figure 23: Implementation Diagram

Adapted from (FatSecret, 2019; Firebase, *n.d.*; Fitbit, Inc., 2018; Misfit, *n.d.*; Realm, 2019; ASICS Digital, Inc., 2018; Under Armour, 2019; Guy, 2015)

5.2.1 GetHealth

We decided to use GetHealth to simplify our interaction with the APIs we supported. GetHealth is a Representational State Transfer (REST) API that is easy to use and supports a variety of third-party APIs that are relevant for our project, including Fitbit, FatSecret, and more.

Table 4 shows all the APIs that HerdHealth supported through GetHealth and the data points which could be pulled for each API. We integrated with GetHealth’s API by creating a GetHealth user for each HerdHealth user account, then requesting authentication from each of the APIs for the devices and applications the users wanted to connect to (GetHealth, *n.d.*). Once an API was connected, the health and fitness data supported by the API could then be retrieved and displayed on the user’s home screen in the HerdHealth.

	Step count	Calories Burned	Distance Traveled	Calories Consumed	Fat	Protein	Sodium
Fitbit	X	X	X	X	X	X	X
FatSecret				X	X	X	X
MapMyWalk	X		X				
Misfit	X	X	X				
RunKeeper		X	X				

Table 5: Applications Supported by GetHealth and Associated Data Types (GetHealth, *n.d.*)

Originally, before we decided to use GetHealth, we planned to use Shimmer, an application provided by Open mHealth that facilitates pulling data from third party APIs and converting it to a format compliant with Open mHealth’s schemas. Once a user has been granted access to a supported third-party API through registration, Shimmer can connect with the third party user account, make requests to the API, and return data either in its original form or in a form normalized to fit Open mHealth’s schemas. While we were able to successfully connect Shimmer to Fitbit’s API, we found that doing so required the use of a Docker container on a

personal computer. Since Docker containers could be used within an Android application, we sought out alternative solutions and ultimately decided to use GetHealth to integrate the data.

Throughout our development and testing, however, the team ran into a few issues with GetHealth. We noticed that data was only displayed if it was entered on the day that it occurred. Therefore, if a user forgot to enter the food that he or she ate on Sunday, but went back on Monday to enter it in the FatSecret app under Sunday's meals, GetHealth would not pick up on this change. Since GetHealth was our primary source for health data, our application's nutrition data did not match up exactly with what was shown in FatSecret. A simple solution to this problem would have been for the user always input the current day's data before the end of the day, but an intuitive application cannot expect the user to be perfect in this way.

Another issue we noticed with GetHealth was that it did not retrieve all data types that we believed to be available from the various APIs. Specifically, it was supposed to pull the number of flights of stairs (i.e. floors) climbed by the user in a given day, but it always returned "0" for this field. Similarly, it always returned "0" for the number of carbohydrates (measured in grams) consumed by the user when pulling nutrition data from FatSecret. We inferred that GetHealth was written a couple years prior to this project and had not been significantly edited since then. For this reason, we think that the various APIs could have changed slightly without GetHealth being updated, so this could cause a disconnect between GetHealth and the APIs. This could be the reason GetHealth did not retrieve all available data types correctly. To avoid this problem, we simply did not include the missing fields in our application, but they could easily be added back if the problem were to be solved in the future.

Unfortunately, we could not fix either of these problems because we did not have access to the GetHealth source code. Alternatively, our application could have circumvented GetHealth

altogether by communicating directly with the APIs to retrieve the missing data, but this approach would have defeated the main purpose of using GetHealth because we would have had to do this for every API we support. The issue where some fields returned “0” did not significantly affect our app because we simply didn’t include those fields in the final version. However, the issue that caused some data to be incomplete impacted the major components of HerdHealth. Beyond the home screen, where the data was displayed, both the individual and community data could have been incorrect on the statistics page. Additionally, the flawed data could have affected the leaderboard, which may have caused HerdHealth to mistakenly add people to, or remove people from, the leaderboards when it shouldn’t have.

### 5.2.2 Google Firebase

Google Firebase is a backend service designed to make writing mobile and web applications as easy as possible. Similar to Amazon Web Services (AWS), Firebase provides authentication services, a responsive database, cloud storage, and more. We used its authentication API for administrative purposes, such as signing up and logging in users. Firebase provided the Realtime Database, a NoSQL database stored in the JSON format, which held all of HerdHealth’s data (Firebase, *n.d.*). The database was split in four JSON branches: fitness data, nutrition data, goals, and users. Fitness data included fitness information, such as steps and calories burned, for each user for each day. Nutrition data included the same, but for nutrition information, such as calories consumed. Goals included each user’s fitness goals for each of the data points, and users included basic user information, such as name, graduation year, and a list of APIs to which the user was connected. Firebase’s cloud file storage was used for any files we

did not need to store locally, such as profile pictures. A diagram of our Firebase database is shown in Figure 24.

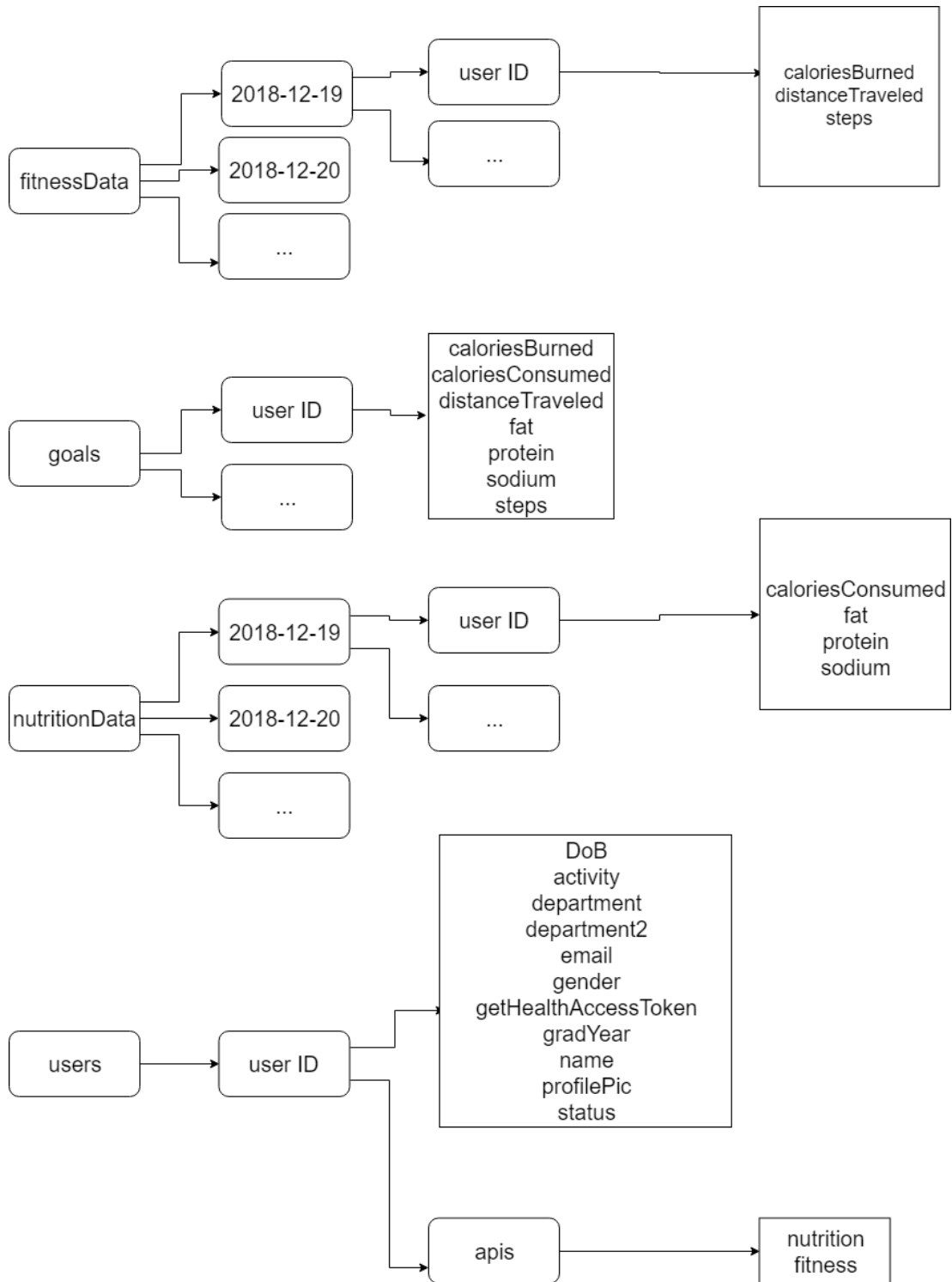


Figure 24: Diagram of Database in Firebase

We chose Firebase because it is very responsive, easy to set up, and it takes advantage of database listeners to alert the application when any data is changed, which minimizes the number of API calls we had to make. In addition, due to our relatively small user base, Firebase was free to use for our purposes because we did not reach the minimum threshold of users or authentication instances at which Google begins to charge developers. We saw Firebase as a good backend service for us to use because it was simple to implement in our relatively small application, but is capable of scaling up to support millions of users. This is one reason why large organizations trust Firebase to support their applications, such as Lyft, Venmo, and Shazam (Firebase, *n.d.*).

### 5.2.3 *Realm*

The team chose to use the third-party database library, Realm, to help us store data locally. Realm is an open source third-party library that acts as a persistent replacement for traditional methods of storing data locally in mobile applications, such as SQLite on Android and Core Data on iOS. Realm requires very little setup and virtually no boilerplate code, which is one advantage over using SQLite. Realm operations use flexible method calls rather than SQL code, making queries simple and readable. Operations are made efficiently, even with large data sets, so Realm is trusted by some of the world's largest technology companies, such as Google, Amazon, and IBM (Realm, 2019). Each table in Realm is based off of a Java class which makes saving custom classes in Realm simple. Realm only requires the classes to “extend” its *RealmObject* class, include an empty constructor, and include getters and setters for all fields.

We used Realm to store most of the information from Firebase that is pertinent to the user, including profile information, the APIs to which the user was connected, fitness goals, and

summaries of fitness and nutrition data. This helped us minimize the number of calls we had to make to Firebase, which was important because, while Firebase is fast, it still takes more time than reading data out of local memory. Also, Firebase calls require an Internet connection, whereas accessing locally stored data does not, so Realm allowed us to display saved data to the user when no Internet connection is available. Due to its efficiency, scalability, and ease of setup and use, Realm was a good method of storing data locally for our application.

#### 5.2.4 *OkHttp*

In order to communicate with GetHealth, we needed to make a few HTTP requests to GetHealth's REST API. For example, creating a new GetHealth user, getting the latest health data, and retrieving the APIs to which a user was connected all required HTTP requests. In order to make these, we used OkHttp, an open source HTTP client designed to make efficient HTTP calls specifically for Android and Java applications (OkHttp, *n.d.*).

#### 5.2.5 *Gson*

Gson is a JSON encoder and parser developed by Google (Google, 2008). We utilized Gson to easily parse the JSON responses retrieved from GetHealth via OkHttp. One feature of Gson that was especially helpful was its ability to parse data directly into a custom Java object. For instance, if a JSON response had the structure shown in Figure 25, Gson would be able to easily create a new *SampleClass* object shown in Figure 26. We made use of this feature when retrieving fitness and nutrition data from GetHealth because it did in one line of code what otherwise would have taken several.

```
data: {  
  "field1": "value1",  
  "field2": "value2",  
  "field3": value3  
}
```

Figure 25: Sample JSON Response

```
public class SampleClass {  
  private String field1;  
  private String field2;  
  private int field3;  
  
  public SampleClass(String field1, String field2, int field3) {  
    this.field1 = field1;  
    this.field2 = field2;  
    this.field3 = field3;  
  }  
}
```

Figure 26: Sample Java Class

### 5.2.6 *GraphView*

*GraphView* is an open source library made specifically for displaying graphs in Android applications. The library supports several different types of graphs and allows for extensive customization (Gehring, 2019). We used *GraphView* to create bar graphs on the statistics page, which compare the user's individual daily statistics to those of the WPI community as a whole.

### 5.2.7 *Glide*

*Glide* is another open source library made specifically for Android, which helps developers efficiently load and display images. Since all of the user profile pictures are located in Firebase Storage, they must be downloaded before they can be displayed in the *ImageViews*. A conventional approach might be to download the images from Firebase, save them locally to the



phone, and then display them. However, this can take a few seconds and use unnecessary resources, such as storage space, battery, and data. Glide expedites this process through caching and algorithms that minimize the storage needed for each image. Whenever we load an image into an *ImageView*, whether local or stored in Firebase, we used Glide. Glide also offers many useful features, such as circle-crop, which crops the image specifically for a round *ImageView*. In addition, we make use of its placeholder option, which puts a default image in the *ImageView* while the real image is loading. If the load fails for some reason, such as an invalid URL, the default image simply remains in the *ImageView* so the user is never left wondering why there is a blank view on the screen (About Glide, *n.d.*). Whenever we save a user profile picture in Realm, we actually just save a URL pointing to the user's profile picture in Firebase, and then have Glide retrieve the image for us.

### **5.3 Development Decisions**

There were a few decisions the team made throughout the development process that might not be obvious at first glance. Here, we will explain the decisions we made, why they are important, and how we arrived at them.

#### *5.3.1 GetHealth Sync Google Cloud Function*

GetHealth had two synchronization functions that updated GetHealth's database with the most recent data from the connected APIs: one to sync all connected users associated with our app with all of their connected APIs, and one to sync a single user with all of his or her APIs. After hitting the "sync all" endpoint, the synchronization would take up to a minute because it made numerous calls of its own, while the "single user" endpoint usually took no more than 15

seconds. When a user connected an API to HerdHealth, it called the “single user” sync so the user’s data would show up on the home screen right away. GetHealth would run the “sync all” function on its own, but their documentation was not clear as to exactly when, or how many times per day, it would run.

Google Cloud functions are JavaScript functions that can be triggered by a number of events, such as changes in specific locations of the Realtime Database, which are run in the cloud to reduce the amount of computation needed on the client-side. In an effort to keep our data more up-to-date with the API databases and improve user experience, we created a Google Cloud function that hits the “sync all” endpoint every 6 hours. Google provides \$300 in free credits to new Google Cloud users, so we were able to schedule this job without incurring any costs to the team. We had it set to run every 6 hours because we didn’t know how quickly we would accumulate Google Cloud costs, but even that might not be often enough for optimal user experience. Therefore, further research could be done on the best balance between up-to-date information and manageable Google Cloud costs.

### *5.3.2 Selectively Retrieving GetHealth Data from Application*

Our application contained two asynchronous tasks that hit GetHealth endpoints on background threads with GET requests to retrieve the current user’s health data; one is for fitness data and one for nutrition data. They were each set to retrieve data from the previous two weeks and save it to both the Realtime Database and Realm. The tasks were called on the home page if the user had at least one API connected for the respective data types. It was important for us to refrain from calling the tasks every time the user navigated to the home page because they each

took a few seconds to run and display the data, which could have led to sluggish loading and a poor user experience.

To alleviate this issue, we set HerdHealth to keep track of the last time it retrieved fitness and nutrition data from GetHealth. If the user navigated to the home page within one hour of the last sync, the data was simply read from Realm to improve the user experience through shorter loading times. On the other hand, if the last sync was more than one hour ago and the user was on the home page, the app would retrieve the newest data from GetHealth, display it, save it to both databases, and update the time of the most recent data synchronization. The app kept track of both the last nutrition data sync and the last fitness data sync separately, to avoid both tsaks being called when only one data type needed synchronization. This check reduced the number of HTTP requests the app had to make, which improved user experience by increasing responsiveness and decreasing battery and data usage. There was also a “Sync Data” option in the home page’s options menu that would retrieve the most recent data regardless of the time of the last sync, so the user would never have to wait to see the latest available data.

### *5.3.3 Nutrition Goal Achievement Streaks*

The idea of the leaderboard page was to recognize the people who were leading the community in different health metrics in an attempt to motivate the other community members to improve their own health metrics. For fitness data, displaying the top users’ data was intuitive because generally, the higher the data is, the better. For example, increasing steps taken, calories burned, and distance walked or ran are all positive behaviors. Therefore, it was easy to show leaderboards for fitness data; we simply showed the highest data values for each fitness metric in the last seven days.

When it came to showing a nutrition leaderboard, however, things got a little trickier. This was because having the highest or lowest nutrition numbers is not always healthy; therefore, it is not something we wanted to encourage. For instance, if someone consumes 10,000 calories one day and someone else consumes 500 calories, we did not want to show either of these users at the top of the nutrition leaderboard because neither would be considered healthy habits for most individuals. Furthermore, recommended caloric intake depends heavily on factors such as gender, height and weight, so it is not realistic to hold everyone to the same standards. Our solution was to analyze each user's nutritional data and give them credit when they were within 10% of all of their intake goals. Thus, the user who achieved (or got within 10% of achieving) all of his or her goals the most days in a row would be at the top of the nutrition leaderboard. It is important to note that only "active" streaks showed up on this leaderboard, where "active" means that the streak had yet to be broken. It may not have been obvious at first why the data on the nutrition leaderboard was measured in days, but that was due to the difficulty that comes with comparing nutrition data among people with different nutritional needs. There was an information button on the action bar of the leaderboard page that opened a dialog which explains this concept, so the understood why we chose to display the data in this way.

#### **5.4 Custom Java Classes and Interfaces**

While developing HerdHealth, we created several custom Java classes and two interfaces to organize and encapsulate our code. In this section, we will explain what each of them do and how we used them.

### 5.4.1 Custom Classes in Realm

We created many of our classes with the intention of storing data in our Realm database because we knew they would be frequently accessed. Figure 27 shows the fields of each of these classes.

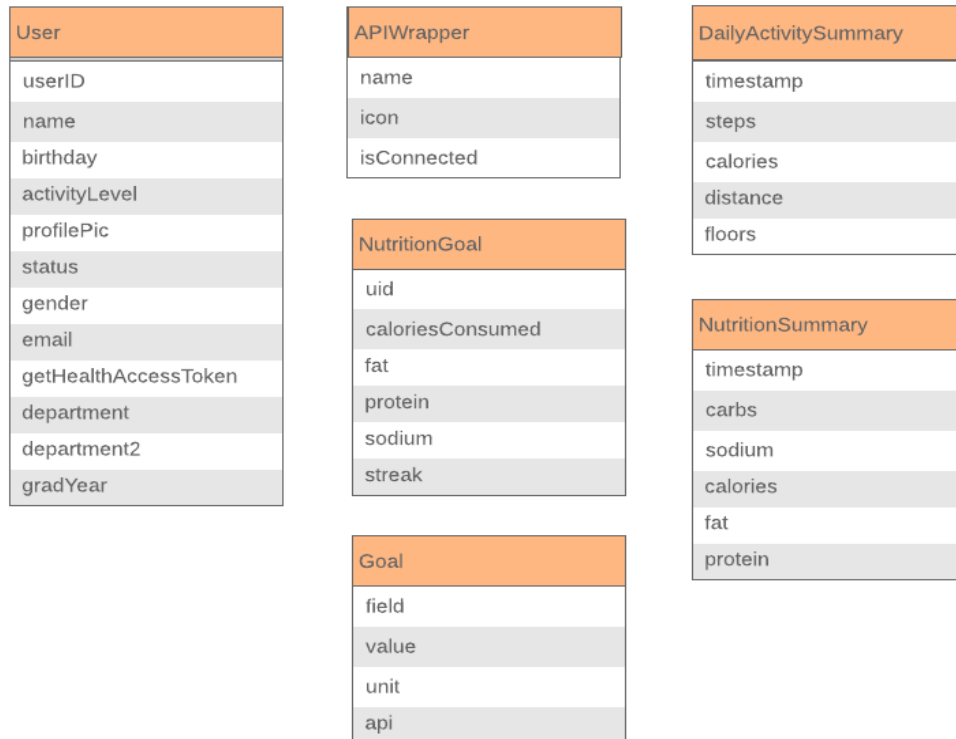


Figure 27: Diagram of Classes Saved in Realm

#### User

The *User* class was used to save all user information that was input on the sign up screen. It is important to note that a user's information was only stored on his or her own device. This class is simply used to keep track of basic information that didn't change often in order to minimize the number of calls to the Realtime Database. This class also saved two values that were needed for user identification. In order to keep track of users, Firebase assigns each authenticated user a 28 character alphanumeric user identification code, or "uid." This code was

saved under the “userID” field in our User class. The second value for identification was the “getHealthAccessToken.” Since GetHealth also uses an alphanumeric code (this one is 40 characters) to identify users, we saved this code, which GetHealth calls an “access token,” in the User class as well. This token was used in each GetHealth call for retrieving data so GetHealth knew which user’s data to pull. When new users signed up for our app, they got a new GetHealth access token assigned to them.

### **APIWrapper**

The *APIWrapper* class was mostly used to keep track of which APIs the user has connected at a given time. The class held the name of each API with a boolean flag denoting the APIs connection status (true meaning the API was connected, false meaning it wasn’t). In addition, it held an integer ID for the app’s logo. Each image resource associated with our project had an assigned ID number, so that was how we chose to represent the app icons in this class.

### **Goal**

This class stored a user’s goals for each type of health data we supported. The field represented the type of data (e.g. steps, calories consumed, etc.) and the value represented the user’s goal for the corresponding field. Each field had a unit that was automatically set based on the field given in the *Goal* constructor. Finally, we also saved the API from which the user wanted to pull data for each field, which was referenced when it came time to fetch the most recent data.

### **NutritionGoal**

The *Goal* class was useful for keeping track of singular goals for the signed-in user, but for the nutrition leaderboard page, we needed a way to keep track of the nutrition goal achievement streaks for all users. To achieve this, we created the *NutritionGoal* class and saved

it in Realm. Each row was associated with a different user and stored information about their goals for all nutrition-related fields as well as how many days in a row their nutritional data was within 10% of their goals. The streak column only contained “active” streaks, so not coming within 10% of all nutrition goals for a day would reset the streak to 0. This class made it easy to find the users with the longest active streaks, as we were able to query this class in Realm and sort the results by the streak column in descending order to get the leaders for the leaderboard page.

### **DailyActivitySummary**

This class stored information about a user’s fitness data that was retrieved from GetHealth. This was done through GET requests, which specified the API from which to retrieve data and the user’s GetHealth access token. When the request was made, a JSON response would be returned. We then used the Gson library to create *DailyActivitySummary* objects based on the returned JSON data. These objects were immediately saved to both the Realtime Database and Realm so we didn’t have to make the relatively slow call to GetHealth everytime we want the user’s fitness data. We displayed the data saved in these objects on the home and statistics pages.

### **NutritionSummary**

The *NutritionSummary* class was the equivalent of the *DailyActivitySummary* class for nutrition data. It saved information retrieved from GetHealth, such as fat, protein, and calories consumed, for each day. It used the same process of making a GET request to GetHealth and using Gson to parse out the response and create *NutritionSummary* objects.

One key difference between this class and the *DailyActivitySummary* was due to the way GetHealth returned nutrition data. GetHealth split up nutrition data by item eaten. For example, if we made a GetHealth call for one day and the user logged 10 items that he or she ate that day,

the response would contain an array of all 10 items with their respective nutritional information. All food items belonging to the same day would have the same timestamp, representing that day at midnight. If we followed the exact same process that was used for the *DailyActivitySummary* Gson would have created 10 different *NutritionSummary* objects, each containing the nutrition data of one food item logged for that day. Therefore, the *NutritionSummary* class contained a function called “mergeData” that combined any *NutritionSummary* objects with the same timestamp, summing the corresponding fields. This allowed us to reduce the 10 *NutritionSummary* objects into a single *NutritionSummary* for the entire day that would be saved to the Realtime Database and Realm.

#### 5.4.2 Custom Classes Not in Realm

While we thought it was worthwhile to store many of our classes in Realm, we did not save all of them locally. Figure 28 shows the classes that were important to our application, but were not saved to our local database. These classes will be described in more detail in this section.



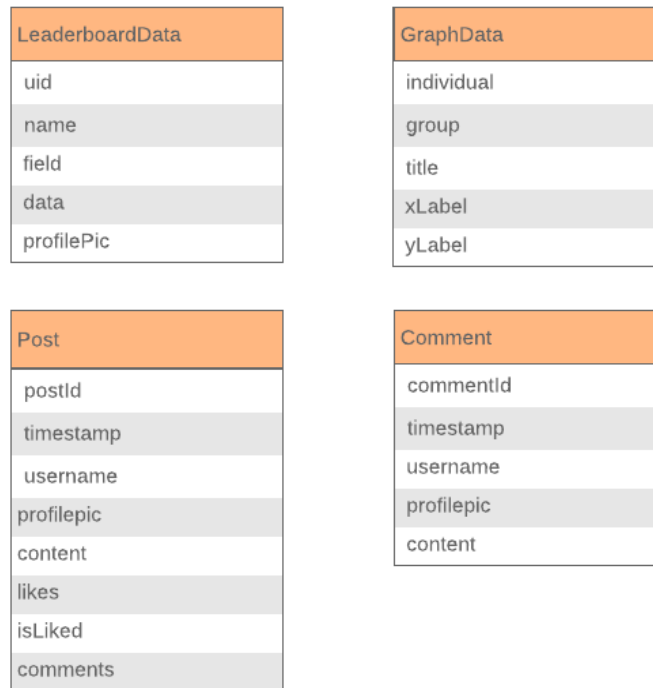


Figure 28: Custom Classes Not Implemented in Realm

## LeaderboardData

The *LeaderboardData* class contained information needed to display each row of the leaderboard, such as the data field (e.g. steps), value (e.g. number of steps), and name of the user whose data was being displayed. The data value was stored as an *Object* (rather than the more specific *int* or *double*) to accommodate all types of data, and this class implemented the *Comparable* interface so *LeaderboardData* objects could be compared based on this value, which allowed for sorting and displaying in descending order. *LeaderboardData* objects not stored in Realm because the Realtime Database queries were adequately fast. However, if the number of users were to grow into the hundreds or thousands, it might behoove future developers to store this class in Realm to reduce loading time because the Firebase queries will take longer. When navigating to one of the leaderboard pages, the data would then only be updated if it

hadn't been updated in a while, similar to the way we selectively synced with GetHealth on the home page based on the last sync time.

## **GraphData**

This class stored all the data needed to display a graph on the statistics page, including both the user's and community's data from the past week, the graph's title, and axis labels. The daily data for both the user and community were of the *BarGraphSeries* type, which is a class provided by the GraphView library. When we wanted to display the data, we simply called the "*GraphView.addSeries*" function twice: once passing it the user's data and once with the community's data. Creating this class made it easy to create and display several different graphs at once through the use of a *RecyclerView.Adapter* object.

## **Post**

The *Post* class stored relevant information needed to display posts on the feed page, such as a timestamp representing when the post was uploaded, the name of the user who posted it, and the content of the post. The feed page created two sample *Post* objects that the user could like and comment on. Likes and comments were not saved, so they did not persist after the user left the feed page. The class was designed to visually represent data pulled from Firebase, should the feed page be fully implemented in the future. Since the feed page was not fully implemented, there were some fields in the *Post* class that did not end up being used, such as an identification number and a boolean flag representing whether or not the signed-in user had liked the post. In a complete feed page, posts would be stored in order from the most recent to least recent timestamp, and new *Post* objects would be created after pulling data from Firebase.

## Comment

The *Comment* class stored information needed to display comments on the feed page, such as the timestamp representing when the comment was created, the name of the user who created it, and the content of the comment. Similar to the *Post* class, the *Comment* class was created with the intention of visualizing data from Firebase on a feed page to be implemented in the future. Each *Post* object had a list of *Comment* objects associated with it. This functionality was present in the two sample *Post* objects that were shown by default on the feed page, but as the objects were not saved, the lists of *Comment* objects would reset when the user navigated to another page.

### 5.4.3 Custom Interfaces

To provide a standardized way of storing data and completing various tasks, we created two Java interfaces. Establishing standardized practices made our application easily mutable and scalable. The code for our interfaces can be found in Figure 29 and Figure 30, respectively.

```
import java.util.TreeMap;
public interface IDataType {
    int numFields();
    TreeMap<String, Object> getData();
    void addToRealm();
    void addToFirebase(String uid);
}
```

Figure 29: IDataType Interface Implementation

```
public interface ITaskFinished {
    void onITaskFinish();
}
```

Figure 30: ITaskFinished Interface Implementation

## **IDataType**

Classes that implemented the *IDataType* interface were meant for saving and performing operations on the different types of data supported. HerdHealth supported daily fitness data (e.g. steps, calories burned) and nutrition data (e.g. fat, calories consumed) so both the *DailyActivitySummary* and *NutritionSummary* classes implemented the *IDataType* interface. This interface contained four methods. The first method simply returned the number of fields belonging to the data type. We supported three data fields for *DailyActivitySummary* (caloriesBurned, distance, and steps) and four fields for *NutritionSummary* (caloriesConsumed, fat, protein, and sodium). This number was used on the home page and statistics pages to define how many rows were needed to show all the data fields we had for each data type.

The interface also had a function to get all the data fields, which simply returned a *TreeMap* with *String* keys and *Object* values. The key was a field, such as “steps,” while the value would be the number of steps. The value was of the *Object* type because some data values were retrieved from GetHealth as *double* values while other were integers (Java primitive type *int*). Therefore, we used the generic *Object* class to accommodate all data types. A *TreeMap* was used here, rather than the more common *HashMap*, because of the *TreeMap* class’s unique ability to sort its keys alphabetically, which was useful for displaying the data in an order that made sense. The other two functions were simply used to save the data pulled from GetHealth. One saved to the Realtime Database and the other saved to Realm. We designed the *IDataType* interface to keep our code clean and make it easy to add new data types in the future. For example, if future developers want to expand HerdHealth to support sleep data, they could start by creating a class for sleep that implements the *IDataType* interface.

## **ITaskFinished**

HerdHealth utilized a few asynchronous tasks to run operations, mainly HTTP requests, on a background thread, which is the standard in Android programming for potentially slow processes. In some cases, we needed to know when an asynchronous task was finished because one or more function calls relied on the task being complete. For this reason, we created the *ITaskFinished* interface, which simply has one function: “onITaskFinish.” When we needed to be notified of task’s completion, we passed an *ITaskFinished* instance into the task constructor and called the “onITaskFinish” function when the asynchronous task was complete. In each respective “onITaskFinish” function definition, we called the function(s) waiting to run. One example of this interface being used was in the asynchronous tasks where we pulled data from GetHealth. In these background tasks, we made a call to GetHealth, parsed the JSON response into the respective class that implemented *IDataType*, then saved the objects to both Realm and the Realtime Database. In these cases, the data needed to be saved to Realm before it could be displayed, so we used the *ITaskFinished* interface to alert the calling class when all the data had been saved and was ready to be displayed.

## 6. Evaluation

We evaluated HerdHealth in two phases:

- *Phase 1:* We showed the application to a few friends, let them navigate through the different pages, and took notes on some of their comments. This was a quick first step which allowed us the opportunity to make any small fixes that could be coded quickly.
- *Phase 2:* We recruited 23 beta testers to evaluate HerdHealth and fill out a Qualtrics survey so we could get some feedback.

Of the 23 beta testers, 13 used at least one of the applications that we support. We specifically sought users of the applications we support so they could see their own data while testing out HerdHealth because we thought that this would be more meaningful than having them look at someone else's data. Since we couldn't recruit enough people who used the apps we support, we then sought out athletes and people interested in physical activity because they are potential users of HerdHealth. Before starting the test, we gave each beta tester a quick rundown of all the features they could expect from our application, but did not include information on how to use, interpret, or navigate to the features. We chose this method because we thought it would give us insight into the usability of HerdHealth.

The survey that the testers filled out after evaluating the app employed the System Usability Scale (SUS), a verified method of measuring the usability of a system (System Usability Scale, *n.d.*). The first ten questions of our survey were those used in the SUS. The survey also asked about the user's most and least favorite features of the app, as well as what they would like to see from the app in the future. Finally, the survey ended with a few demographics questions so we could better understand our users. The complete list of survey questions, including those from the SUS, can be found in Appendix 2.

## 7. Results

This section will outline the results of our evaluation sessions, including survey results and observations we made while watching the beta testers navigate through our app.

### 7.1 Evaluation Survey Results

We were able to accumulate 23 responses to our evaluation survey: 1 from each of our beta testers. Unlike the first survey we conducted to gain knowledge of initial app interest, the evaluation survey used no flow logic to customize respondents' questions based on their answers to previous questions. Since everyone had the same survey experience and no one opted to skip questions, all 23 participants answered all the questions.

#### 7.1.1 Demographics

Of the 23 WPI community members we surveyed, 87.0% (20) were students, 8.7% (2) were faculty, and 4.3% (1) were staff. This distribution was shown in Figure 31 below.

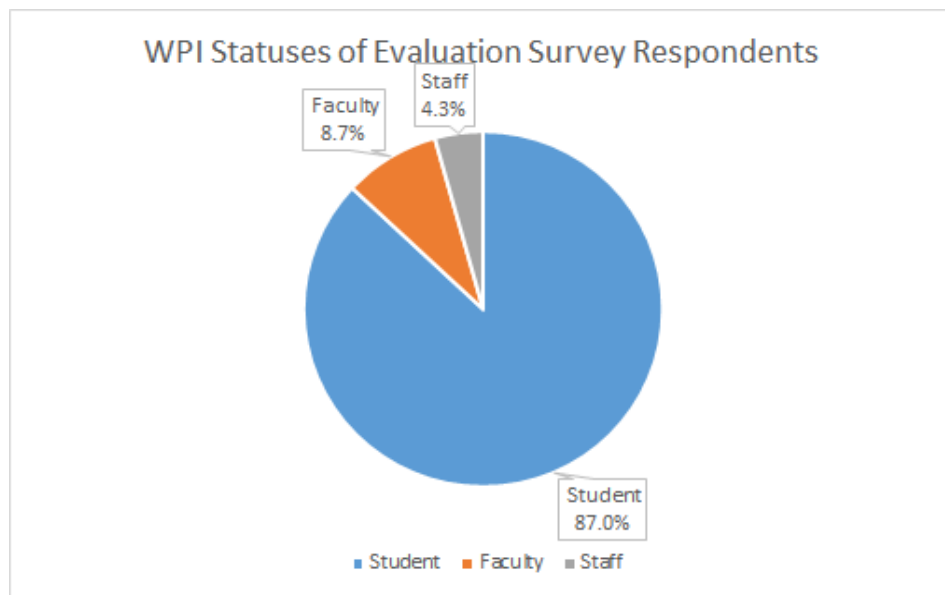


Figure 31: WPI Status of Evaluation Survey Respondents

Figure 32 showed the breakdown of respondent genders. 56.5% (13) of the respondents were male, while 43.5% (10) were female.

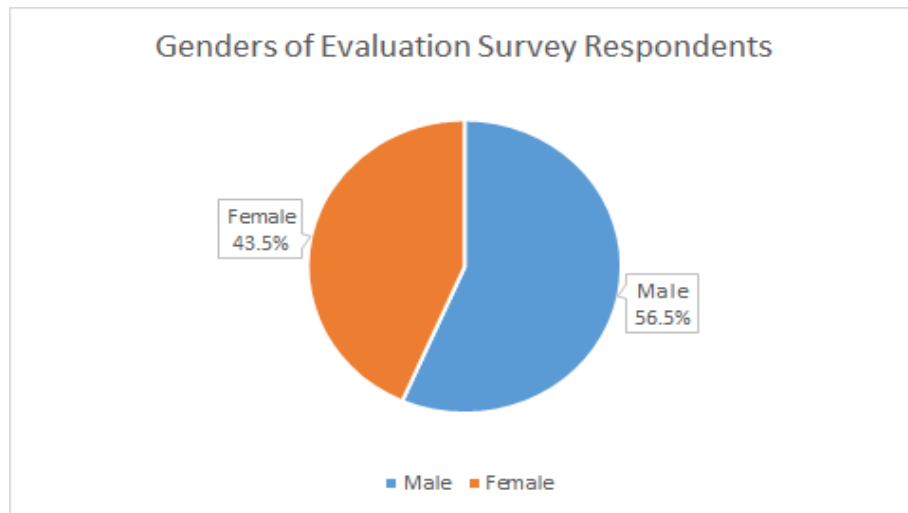


Figure 32: Gender of Evaluation Survey Respondents

Figure 33 showed the ages of the evaluation survey respondents. Since the majority of participants were students, most people (87.0%) fell in the 18-22 age range. The three faculty and staff members were 28, 36, and 45 years of age respectively.

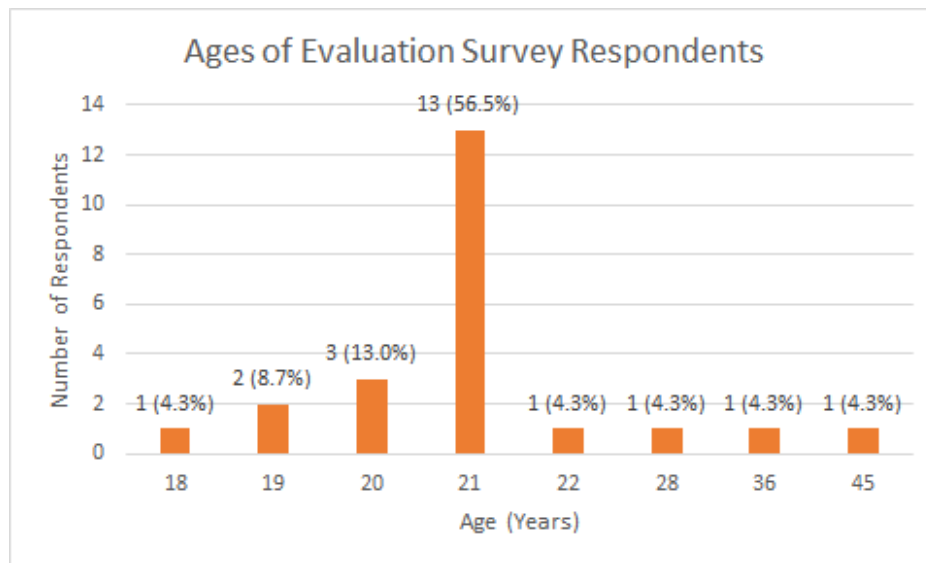


Figure 33: Ages of Evaluation Survey Respondents



Figure 34 showed the reported activity levels of our beta testers. There were 26.1% (6) who reported being “Lightly Active” and “Moderately Active,” respectively, while the other 47.8% (12) said they were “Active.”

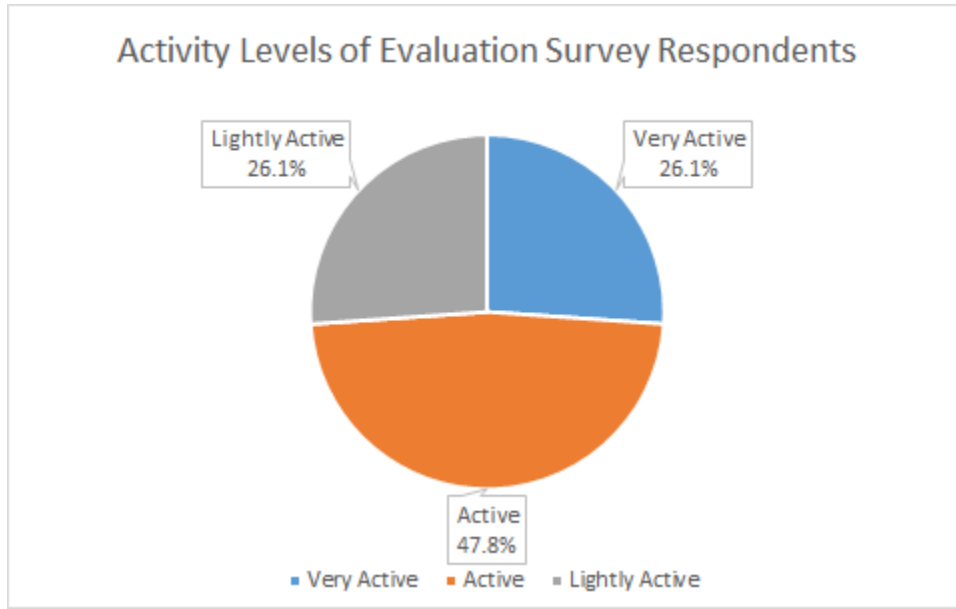


Figure 34: Activity Levels of Evaluation Survey Respondents

The last demographics question we asked participants was what type of phone they had. 78.3% (18) of the participants owned an iPhone, while the other 21.7% (5) owned an Android. Figure 35 below shows this distribution.

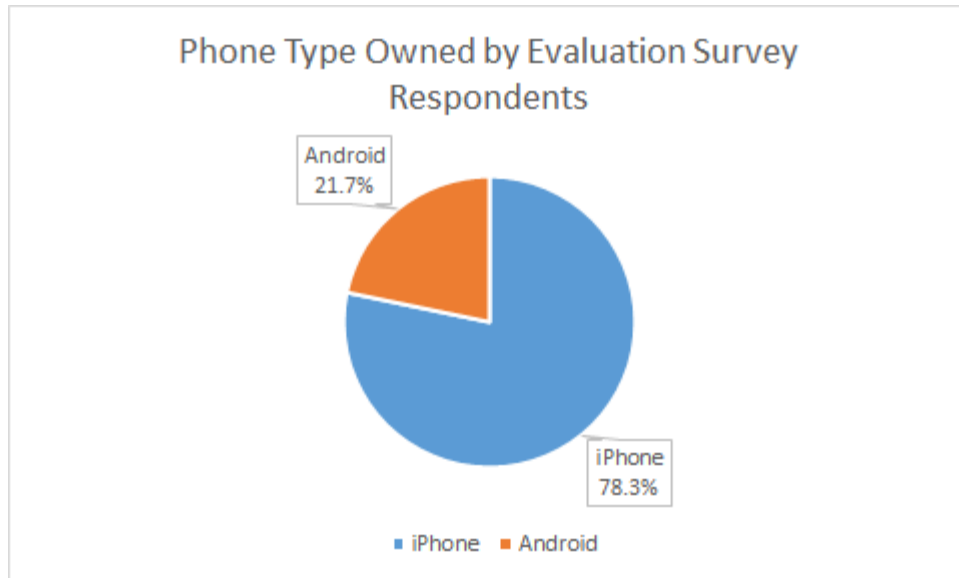


Figure 35: Phone Type Owned by Evaluation Survey Respondents

### 7.1.2 System Usability Scale

The full SUS results and scoring details could be found in Appendix 3. Using the scoring instructions, we calculated a score of 82.83 out of 100 for HerdHealth. While the score was out of 100, it was not a percentage and should not be interpreted as such. Any score above 80.3 was considered an “A” on the SUS (Thomas, 2015), so our beta testers reported that our app had high usability.

### 7.1.3 Most Liked Feature

When asked which feature (or features) beta testers most enjoyed, 34.7% (8) said they most liked the statistics page’s ability to visualize data in graphs and compare their own data to the WPI averages. Additionally, the leaderboard and the fact that the app could retrieve data from multiple sources were each mentioned by 26.1% (6) of the participants. People also expressed that they liked the potential for the groups and feed features, the user interface, and the ease with which they could see how close they were to achieving their goals on the home page.

#### *7.1.4 Least Liked Feature*

17.4% (4) of the participants stated that they were least likely to use the app's ability to track nutrition data, with one saying specifically that it was too time-consuming. Furthermore, 8.7% (2) reported disinterest in both the feed and leaderboard features, respectively. There was a much smaller range of features respondents reported not liking than those they reported liking. 17.4% (4) of the respondents simply said they didn't dislike any of the app's features.

#### *7.1.5 Additional Types of Data Users Want to Track*

We asked the beta testers if there were any additional types of data they would like HerdHealth to support that it at the moment did not so we could better understand what the WPI community was looking for in a mobile health application. 30.4% (7) of the beta testers reported wanting to see how many floors or flights of stairs they climbed. In particular, this might be especially relevant in a hilly city like Worcester. Additionally, 13.0% (3) of the testers said they wanted to be able to track carbohydrates using our app. Among the other responses were sleep (8.7%), active minutes (8.7%), and weight lifting (4.3%).

#### *7.1.6 Suggestions for Application's Appearance*

When asked if respondents had any ideas on how to make the appearance of the application better, 43.4% (10) said they had none. More appealing colors (or better variation of colors) and labels for buttons people didn't notice were each mentioned by 8.7% (2) of the respondents. One respondent reported that each data type should have an app icon next to it on the home screen to show the source of the data (e.g. show the Fitbit icon next to data that came from Fitbit).

### 7.1.7 Features Users Want in the Future

When we asked the users about additional features they'd like the app to have in the future, answers were varied, but there were a few commonalities. 8.7% (2) of the participants said they wanted to be able to input data directly into HerdHealth either because they didn't use other apps to track their health or because they wanted the data they input to be synced with their other health apps. An additional 8.7% (2) said they wanted a challenge feature, where people could challenge themselves to achieve some long-term goal or compete against their friends. Another 8.7% (2) said they wanted HerdHealth to support retrieving data from the Apple Watch. Other responses included being able to post a post-workout picture to the feed page and the ability to form groups to compare progress with friends or sports teams.

Another notable suggestion was to expand the statistics graphs and make them scrollable, so that the user could go back more than one week. With this feature, users could get a better idea of what their trends might look like, and therefore would have a better overall understanding of their fitness and nutrition progress. Similarly, users also wanted to be able to scroll back to see more days on the homepage. For this suggestion, a *DatePicker* might be helpful, so that the user could easily select and look back at days that were weeks or months ago.

Finally, another feature users wanted to see would be to give suggestions for fitness and nutrition goals based off of information users enter about themselves. For exercise goals, HerdHealth would suggest values recommended by healthcare professionals. It could also suggest nutrition goals for either weight loss, maintenance, or weight gain, using an algorithm to calculate the user's caloric and macronutrient intake based on their information, such as height, weight, age and activity level.

## 7.2 Observations from Beta Testing

While the beta testers navigated through HerdHealth, we took notes on places where their expectations did not completely match up with the app's behavior, so we could better align them in the future. The most significant point we noticed was that most users could not find the leaderboard page because the button that navigates to it was located in an ActionBar (top of the screen), as opposed to the rest of the features, which were on toolbar on the bottom of the screen. Figure 36 (A) showed the location of the leaderboard button that most people missed. Similarly, many users didn't click on the back arrow on the home page (which would have displayed their data from the previous day) because they either didn't notice it or thought it was a back button due to its placement near the top-left corner of the screen. Many people also tried clicking on the "Fitness" and "Nutrition" headers on the home screen because it wasn't obvious enough that they were simply headers to tell you what type of data is being displayed. These two components of the home screen can be found in Figure 37 (A & B).



Figure 36: Screenshot of Statistics Screen  
 (A): Button that leads to leaderboard

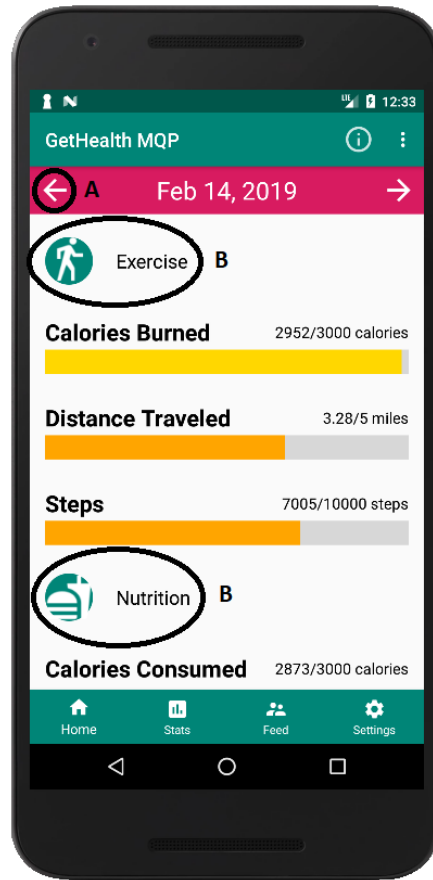


Figure 37: Screenshot of Home Screen  
 (A): Button that shows previous day's data  
 (B): Data headers

Another observation we made was that the app currently only supports WPI students and faculty, not staff. There was no “Staff” option in the dropdown where users select their status at WPI. Users were then asked to pick a major/department, which did not apply to staff members. We also noticed that some users tried to edit their profile information, such as name, department, etc., but they did not click the “Update Info” button at the bottom of the screen to save their changes before navigating away from the page. Figure 38 (A) below shows the button at the bottom of the profile screen that many people missed.

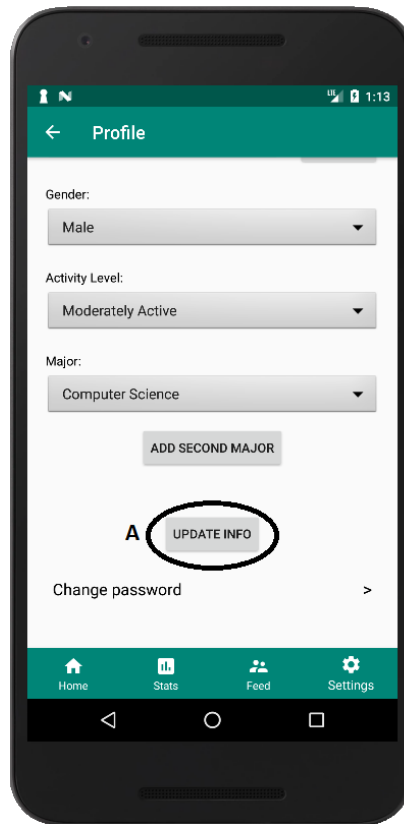


Figure 38: Screenshot of Profile Page  
(A): Update Info button

## **8. Discussion**

In this chapter, we will state our findings from the results of our evaluation and survey, point out potential survey biases and inconsistencies, provide possible usability improvements, and discuss some limitations of our evaluation.

### **8.1 Discussion of Evaluation Survey Results**

In this section, we will analyze the results gathered from the evaluation survey and make note of some interesting findings. It is important to note that any findings or patterns we discuss in this section were made from a relatively small data set (N=23). Therefore, we did not form conclusions based upon the trends we discuss, nor should these correlations be considered causation.

#### *8.1.1 Evaluation Survey Findings*

After analyzing the data, we found a correlation between participants' responses about their activity level and their responses regarding how likely they would be to use the app. As mentioned in section 7.1.1, 26.1% (6) of the survey participants said they were "Lightly Active," while the rest reported being either "Active" or "Very Active." Assigning a numerical value to each of the SUS responses on a scale of 1-5 (1 being "Strongly Disagree" and 5 being "Strongly Agree") allowed us to quantify how much the average person agrees with a particular statement. Using this technique on the "I think that I would like to use the system frequently" statement, we calculated an average value of 2.83 for people who reported being "Lightly Active." On the other hand, those who reported being either "Active" or "Very Active" gave an average value of 4.12, which indicates that our app may appeal more to the more active users than less active users.

Figure 39 below shows the contrast in responses for how frequently they would use HerdHealth



based on their activity level. This could have been because physically active users are more likely than less active users to meet or exceed the WPI averages and appear on the leaderboards, which is more encouraging than being below community averages and not appearing on the leaderboards. If this added encouragement motivated people to use HerdHealth frequently and improve their health habits, it could have been because of the “provide general encouragement” BCT, which shows a correlation between positive feedback and behavior change.

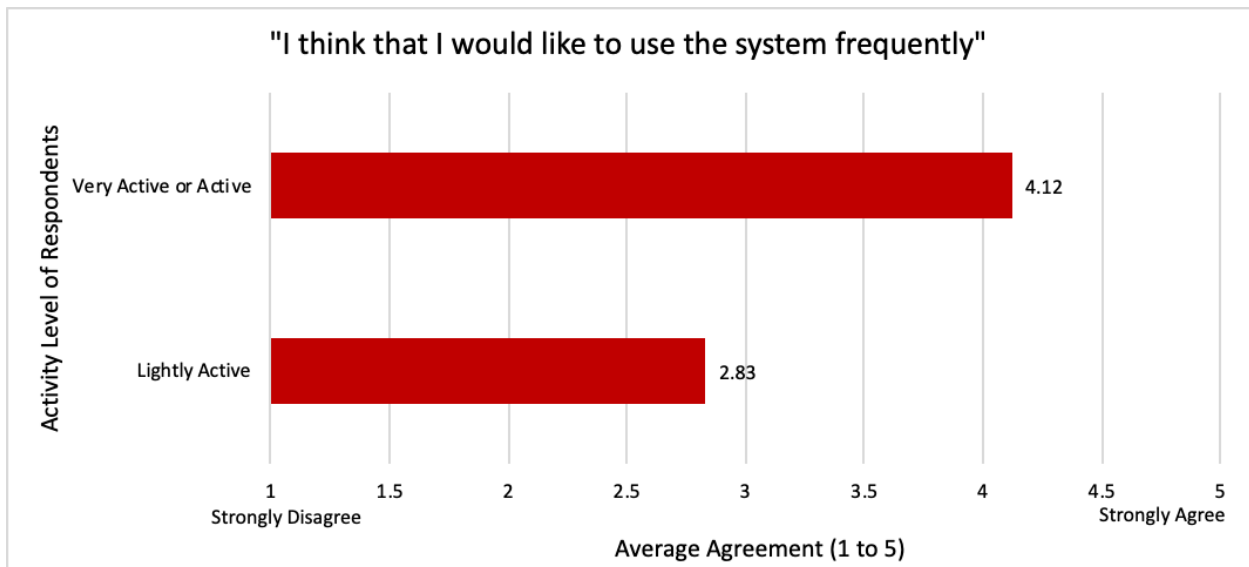


Figure 39: Average Likelihood to Use HerdHealth Frequently, by Activity Level

Another correlation we noticed was a difference in the reported favorite features of males and females. The most common favorite feature mentioned by females was the ability to compare one’s own data to that of the WPI community on the statistics page. 87.5% (7 out of 8) of the people who said the statistics page was their favorite feature were female. Meanwhile, the males’ most commonly reported favorite feature was the ability to compile data from multiple sources in one place. 83.3% (5 out of 6) of the people who said this was their favorite feature

were male. These results suggest that females who use our app may be more motivated by the social comparison BCT applied on the statistics page than males.

### *8.1.2 Potential Evaluation Survey Flaws*

One possible bias we noticed in the evaluation survey was in the question asking which types of data users would like to see that are not shown in the app. We gave “floors climbed” and “carbohydrates” as examples for types of data. However, this may have influenced the answers we received to this question because almost half of the respondents put one of these two data types as their answer.

Another potential issue was in regards to the question asking how active the tester is. The possible responses we allow the user to select from were “Very Active,” “Active,” “Lightly Active,” and “Not Active.” However we didn’t define exact parameters for each response is, so there might have been some inconsistency with people’s individual opinions of how active they think they are and what constitutes each level of activity.

## **8.2 Potential Usability Improvements**

As mentioned in the results, responses to the System Usability Scale questions yielded a score of 82.83 out of 100, which is considered an “A.” While this means beta testers considered HerdHealth to be usable overall, there were some aspects of the user interface that confused them, according to their survey responses. These features of the application could potentially be updated to improve the overall user experience.

A potential fix for better usability would be to make the important buttons more noticeable or change the placement of them. One example of this improvement would be to

resolve the issue with people missing the “Update Info” button at the bottom of the profile page, as discussed previously in section 7.2 of the Results chapter. This issue could have been caused by the button not being immediately visible when a user navigated to the page. In order to see the button, the user needed to scroll all the way down to the bottom of the page, since the information on that page was lengthy. Some potential solutions for this problem would be to have the page autosave whenever the user changes any information, move the button to the top ActionBar, or give a warning to the user saying that there are unsaved changes if they try to navigate away from the page without saving first. Any of these options would help prevent the issue of users navigating away from the page without their changes being saved.

Another commonly missed button was the leaderboard button, which was a small icon on the right side of the ActionBar on the statistics page. An improvement for this would be to move the icon and add a label that says “Leaderboard” to the bottom toolbar, so that the user knows it is a button that navigates to another page, as well as what to expect on that page.

Finally, one more usability fix would be for the “Fitness” and “Nutrition” headers on the home screen. Since many people tried clicking on these without realizing they were just headers, a quick fix to this would be to make the sections collapsible. Therefore, when a user clicks on the section header, the data underneath would expand or collapse. This would also be a useful feature if more data types are added in future development of HerdHealth. The ability to expand and collapse each of the sections would allow the user to more easily view their data on the home screen and it would make the page look less busy and overwhelming.

### **8.3 Limitations of Evaluation**

The biggest limitation in terms of evaluation was the number of users we were able to gather for testing. We tried to find users that regularly used the apps and devices our app supports, which turned out to be more difficult than we had originally thought. This was mainly due to us not being able to support as many apps and devices as we had hoped. For example, we found from the initial surveys that many people in the WPI community use MyFitnessPal, so we planned to support that in HerdHealth. However, after several attempts to get access to their developer API, we were not able to do so.

Another limitation was that not everyone we recruited for testing had an Android phone. This could have biased our results in terms of features the tester found useful and how they might navigate through the app. In particular, one of the significant issues we found was that many testers could not find the leaderboard page. This may have been influenced by the fact that most of our testers were iPhone users and typically iPhone apps do not have action buttons at the top of the screen. Therefore, on our Android app, they may not have thought to look at the ActionBar at the top of the screen for more buttons.

## **9. Conclusion**

The vast number of health and fitness applications available has resulted in data fragmentation because not all applications support all the data types users are interested in tracking. This project studied the potential of a community health and fitness application that allows members of the WPI community to aggregate data from multiple apps and devices and compare it to that of the rest of the WPI community. We gauged interest in such an application, created a prototype, and gathered opinions on its usability. Responses to HerdHealth were overall positive; beta testers found it to be usable, and many of them expressed interest in the ability to visualize their data against WPI averages. However, we also found several areas of improvement for our app as well as wrote about additional features that could be included in the future.

The evaluation we conducted was limited, but it was enough to indicate interest in future development of this application. Future development could increase the app's appeal and usability for users, as well as widen the audience with which it is compatible. Furthermore, future studies could reveal insight into the app's potential as an offering for the WPI community.

## **10. Future Work**

With so many different fitness devices and applications in use, as well as the complexity involved in social networking, we did not have time to implement our concept for the application in full. There are many features we planned but did not have time to implement, and we believe these features could be a valuable improvement for a future MQP. Individual features are discussed below:

### **10.1 Support More APIs**

While developing our application, we were limited to the APIs GetHealth supports, and even further because GetHealth could not successfully pull data from all APIs it claims to support. The version of the app we used for evaluation only supported five applications: Fitbit, FatSecret, RunKeeper, Misfit, and MapMyWalk. To accommodate more users, we would like future versions of the application to support more APIs. This would give the statistics and community pages a more complete view of how everyone in the community is doing, in terms of their goals and fitness. Particularly, we think HerdHealth would have wider appeal if it supported MyFitnessPal, which a lot of people reported using in the initial survey. However, since MyFitnessPal's API is private to a closed list of developers, we were not able to support it in our application.

### **10.2 iPhone Version**

In order for the application to be used throughout the WPI community, it would require an iOS version to be written. About 67% of the WPI students and faculty we surveyed reported owning an iPhone, so if only an Android version of the application were available, it would exclude about two-thirds of the community. This would be counterintuitive to the purpose of the

app, which is to allow community members to compare health data regardless of which devices or applications they use.

### **10.3 Data Flexibility**

Currently, the app only displays seven types of data: calories burned, calories consumed, steps taken, distance traveled, sodium intake, fat intake, and protein intake. In future versions, we would like to include more data types, such as carbohydrate intake, sleep data and the number of times a user exercised in the current week. For exercise data in particular, GetHealth provides an activities request, which includes more detailed information about physical activity than the daily summary does alone. We would like to take advantage of this API call to include more data from users' applications.

Additionally, we would like users to have more options when choosing which data is pulled from which APIs. Currently, each user has one app designated for fitness data and one for nutrition data. In the future, we would like users to be able to choose a separate app for each individual data type.

### **10.4 User Groups**

One feature we planned, but did not have time to implement, is the inclusion of groups. There would be two types of groups: default and user-created. Users would be automatically assigned to default groups, which would be based on categories such as major and graduation year, while users could create groups for a specific group of people, such as members of a club, Greek organization, or friend group. Users would use the groups feature on the settings page to search for groups to join, and when they chose one, they would send a request to join it. The

group leader would then either accept or deny that request. Each group would have its own feed page, and users would be able to filter data on the statistics page to compare themselves to members of that specific group.

### **10.5 Statistics Filtering and Data Display**

Currently, the graphs on the statistics page only show fitness and nutrition data of the individual user and the community as a whole. Originally, we planned to allow users to filter the data by categories such as major/department, graduation year, and whether the included users were students or faculty. Additionally, if the previously mentioned groups feature were implemented, then users would be able to view graphs comparing their data to the group averages. Users would ideally be able to filter the leaderboards by these criteria as well. Due to a general lack of data and implementation time, we did not build out the filtering features, but our hope is to allow users to have this option in future versions of the application.

We also believe it would be beneficial to include more detail in the data displays themselves. One possibility is to include box and whisker plots, in which the whiskers were black bars drawn on top of the WPI average and other community bars to indicate the maximum and minimum values of the corresponding set of data. Alternately, they may show the 25th and 75th percentile of the community data, to account for users who do not use the app regularly and record 0 steps for a particular day. To gain a more accurate impression of how they compare to the rest of the community, users could also view the percentile at which they fall.



## 10.6 Friends and Feed Page

While we were able to implement the user interface of the feed page, it does not have any underlying functionality. It only displays two sample posts that users can like and comment on, and any likes and comments made by the user are not stored in the database. The fully implemented version of this page would allow users to view goals updates from their friends as well as messages to any user-created groups of which they are a member. Either users would be able to make posts themselves, or posts would be auto-generated when the user reached a certain milestone, such as 50% of a particular goal. If auto-posting were to be implemented, users would be able to turn it off in settings.

Outside of the feed page itself, it would be essential to build in the friends functionality to complete the community feature. On the friends page, which would be accessible through both the community and settings pages, users would be able to remove current friends in addition to searching the user base to add new friends. Users would be able to send out friend requests, which the recipient could either accept or decline. This feature would also include public profile pages for each user, so that when other users can click the user's name, they can view that individual's statistics. While the search functionality on the friends page has been implemented, users do not yet have a friends list, so no users can be added or removed. In addition to a place to store friends in Firebase, it would be necessary to create a notification system to complete the friends feature. This way, users would be alerted whenever someone sent them a request. This could be achieved using a Google Cloud function that would be triggered each time the friends branch of the Firebase Realtime Database is changed. When triggered, the recipient would be sent some kind of alert (e.g. push notification or email) to notify him/her of a new friend request.

When the request has been accepted or declined, a Google Cloud function could notify the sender of the recipient's response.

### **10.7 Challenges and Other Incentives**

As discussed earlier in this paper, social comparison, as well as other motivating factors outlined in the Behavior Change Techniques, have the potential to drive people to reach a goal. Currently, the only motivation to meet fitness goals that we have implemented is the leaderboard and color change of the progress bars on the home page, which change colors depending on how close a user is to that particular goal. A future version of the application should have a variety of motivating factors. Ideas include “challenges” that allow users to compete with friends or group members, information on the benefits of maintaining a healthy lifestyle, a partner system where two users are tasked with sending each other encouraging messages, and “badges” that appear on a user's profile when he or she has made certain achievements. Additionally, the app could send motivational push notifications. For example, an Apple Watch might tell a user to keep up the good work when they are ahead of their normal activity progress at a particular time of day, or conversely, might tell the user to pick up the pace when they have below-average activity on the day. Small nudges such as these could be a good feature to add in the future.

### **10.8 More Communities**

We believe this idea has applications outside of WPI. To fulfill our project vision to the highest possible degree, the application would be distributed to many different communities. Either a customizable skeleton of the application would be distributed, or the application would include the ability to create multiple communities, similarly to how users would be able to create

groups. While colleges besides WPI could have their own communities, the residents of a particular town or members of a community fitness club may obtain one as well. The overall goal of this application would be to facilitate the comparison of health data among those physically close to each other, regardless of which devices individuals use.

## References

- About glide. Retrieved from <https://bumpstech.github.io/glide/>
- Abraham, C., & Michie, S. (2008). A taxonomy of behavior change techniques used in interventions. *Health Psychology, 27*(3), 379-387. Retrieved from <http://psycnet.apa.org/fulltext/2008-08834-010.pdf>
- Ahn, A. (2017). How we're helping people find quality apps and games on google play. Retrieved from <https://android-developers.googleblog.com/2017/08/how-were-helping-people-find-quality.html>
- Apple Inc. (2019). Data types. Retrieved from [https://developer.apple.com/documentation/healthkit/data\\_types](https://developer.apple.com/documentation/healthkit/data_types)
- ASICS Digital, I. (2018). Runkeeper. Retrieved from <https://runkeeper.com/>
- Asimakopoulos, S., Asimakopoulos, G., & Spillers, F. (2017). *Motivation and user engagement in fitness tracking: Heuristics for mobile healthcare wearables*. Basel: MDPI AG. doi:10.3390/informatics4010005
- Bandura, A. (1986). *Social foundations of thought and action: A social cognitive theory*. Englewood Cliffs, NJ, US: Prentice-Hall, Inc.
- Bill and Melinda Gates Foundation. (2011). Strategy overview. *Nutrition*,
- Carlson, S. A., Fulton, J. E., Pratt, M., Yang, Z., & Adams, E. K. (2015). Inadequate physical activity and health care expenditures in the united states. *Progress in Cardiovascular Diseases, 57*(4), 315. Retrieved from <https://www.ncbi.nlm.nih.gov/pubmed/25559060>
- Carter, M. C., Burley, V. J., Nykjaer, C., & Cade, J. E. (2013). Adherence to a smartphone application for weight loss compared to website and paper diary: Pilot randomized controlled trial. *Journal of Medical Internet Research, 15*(4) doi:10.2196/jmir.2283
- Chen, C., Haddad, D., Selsky, J., Hoffman, J. E., Kravitz, R. L., Estrin, D. E., & Sim, I. (2012). Making sense of mobile health data: An open architecture to improve individual- and population-level health. *Journal of Medical Internet Research, 14*(4), e112. doi:10.2196/jmir.2152
- Clipart Panda. (2017). Doctor clipart: Clipart - doctor. Retrieved from [http://www.clipartpanda.com/clipart\\_images/clipart-doctor-58319530](http://www.clipartpanda.com/clipart_images/clipart-doctor-58319530)
- Consolvo, S., Everitt, K., Smith, I., & Landay, J. A. (2006). Design requirements for technologies that encourage physical activity. Paper presented at the *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Montré#233;al, Qu#233;bec, Canada. 457-466. doi:10.1145/1124772.1124840 Retrieved from <http://doi.acm.org/10.1145/1124772.1124840>
- Database icon. (2019). Retrieved from [https://www.iconfinder.com/icons/115268/database\\_icon](https://www.iconfinder.com/icons/115268/database_icon)

- DiMenichi, B. C., & Tricomi, E. (2015). The power of competition: Effects of social motivation on attention, sustained physical effort, and learning. *Frontiers in Psychology*, 6, 1282. doi:10.3389/fpsyg.2015.01282
- Estrin, D., & Sim, I. (2010). Open mHealth architecture: An engine for health care innovation. *Science*, 330(6005), 759-760. Retrieved from <http://science.sciencemag.org/content/330/6005/759>
- FatSecret. (2019). FatSecret. Retrieved from <https://www.fatsecret.com/Default.aspx>
- Finkelstein, Eric A, Prof|Haaland, Benjamin A, PhD|Bilger, Marcel, PhD|Sahasranaman, Aarti, PhD|Sloan, Robert A, PhD|Nang, Ei Ei Khaing, PhD|Evenson, Kelly R, Prof. (2016). Effectiveness of activity trackers with and without incentives to increase physical activity (TRIPPA): A randomised controlled trial. *Lancet Diabetes & Endocrinology, The*, 4(12), 983-995. doi:10.1016/S2213-8587(16)30284-4
- Firestore. Retrieved from <https://firebase.google.com/>
- Fitbit, I. (2018). Fitbit. Retrieved from <https://www.fitbit.com/home>
- Garmin. (2019). Connect. Retrieved from <https://connect.garmin.com/en-US/>
- Gehring, J. (2019). GraphView - open source graph plotting library for android. Retrieved from <http://www.android-graphview.org/>
- GetHealth. Retrieved from <http://gethealth.io/>
- Ghafourifar, A., & Evans, M. (2018). The machine learning revolution: How artificial intelligence could transform your business. Retrieved from <https://www.forbes.com/sites/allbusiness/2018/10/20/machine-learning-artificial-intelligence-could-transform-business/#27eccdd6c6c3>
- Google. Google fit. Retrieved from <https://www.google.com/fit/>
- Google. (2008). Gson. Retrieved from <https://github.com/google/gson>
- Google. (2018). Top free in android apps. Retrieved from [https://play.google.com/store/apps/collection/topselling\\_free?hl=en](https://play.google.com/store/apps/collection/topselling_free?hl=en)
- Guy, E. (2015, Oct 11,). Death to ugly android screenshots. Retrieved from <https://medium.com/@emmaguy/death-to-ugly-android-screenshots-48ea071d912e>
- Higgins, J. P. (2016). Smartphone applications for patients' health and fitness. *The American Journal of Medicine*, 129(1), 11-19. Retrieved from <https://doi.org/10.1016/j.amjmed.2015.05.038>
- iHealth Labs Inc. (2018). iHealth. Retrieved from <https://ihealthlabs.com/mobile-apps/>
- Japsen, B. (2016). Wearable fitness devices attract more than the young and healthy. Retrieved from <https://www.forbes.com/sites/brucejapsen/2016/07/11/wearable-fitness-devices-attract-more-than-young-healthy/>

- Jiryae, N., Siadat, Z. D., Zamani, A., & Taleban, R. (2015). Comparing of goal setting strategy with group education method to increase physical activity level: A randomized trial. *Journal of Research in Medical Sciences : The Official Journal of Isfahan University of Medical Sciences*, 20(10), 987-993. doi:10.4103/1735-1995.172792
- Kesiraju, L., & Vogels, T. (2017). Health & fitness app users are going the distance with record-high engagement. Retrieved from <http://flurrymobile.tumblr.com/post/165079311062/health-fitness-app-users-are-going-the-distance>
- Kosecki, D. (2017). 13 fitbit community features you can customize for more (or less!) privacy. Retrieved from <https://blog.fitbit.com/fitbit-privacy-settings/>
- Leib Litman, Zohn Rosen, David Spierer, Sarah Weinberger-Litman, Akiva Goldschein, & Jonathan Robinson. (2015). Mobile exercise apps and increased leisure time exercise activity: A moderated mediation analysis of the role of self-efficacy and barriers. *Journal of Medical Internet Research*, 17(8), e195. Retrieved from <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4642397/>
- Michael Mathews. (2017). The student information system dilemma. Retrieved from <https://www.thetambellinigroup.com/the-student-information-system-dilemma-when-fud-meets-fatigue-the-whole-industry-suffers/>
- Miragall, M., Domínguez-Rodríguez, A., Navarro, J., Cebolla, A., & Baños, R. M. (2017). Increasing physical activity through an internet-based motivational intervention supported by pedometers in a sample of sedentary students: A randomised controlled trial. *Psychology & Health*, 33(4), 465-482. doi:10.1080/08870446.2017.1368511
- Misfit. Misfit. Retrieved from <https://misfit.com/>
- nicescene. (2017). Doctor on internet online laptop for telemedicine with long shadow medical icon. Retrieved from <http://media.gettyimages.com/vectors/doctor-on-internet-online-laptop-for-telemedicine-with-long-shadow-vector-id691761024?s=170667a&w=1007>
- OkHttp. Retrieved from <https://square.github.io/okhttp/>
- Open mHealth. (2015a). About. Retrieved from <http://www.openmhealth.org/organization/about/>
- Open mHealth. (2015b). What's possible with open mHealth? Retrieved from <http://www.openmhealth.org/features/features-overview/>
- Open mHealth. (2018). Shimmer. Retrieved from <https://github.com/openmhealth/shimmer>
- President's Council on Sports, Fitness & Nutrition. (2012). Facts & statistics. Retrieved from <https://www.hhs.gov/fitness/resource-center/facts-and-statistics/index.html>
- Realm. (2019). Realm database. Retrieved from <https://realm.io/products/realm-database/>
- Ridgers, N. D., McNarry, M. A., & Mackintosh, K. A. (2016). *Feasibility and effectiveness of using wearable activity trackers in youth: A systematic review*. Canada: JMIR Publications. doi:10.2196/mhealth.6540

Samsung Electronics Co., L. (2018). Samsung health.

Schaffhauser, Dian. (2017). Building real community online with free apps. Retrieved from <https://www.google.com/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=2ahUKEwjB-OLs-sPgAhUH8YMKHenfD-oQjRx6BAgBEAU&url=https%3A%2F%2Fcampustechnology.com%2Farticles%2F2017%2F07%2F05%2Fbuilding-real-community-online-with-free-apps.aspx&psig=AOvVaw2biGMCmoAlvrnWzWib2bPR&ust=1550533850758463>

Schoeppe, S., Alley, S., Van Lippevelde, W., Bray, N. A., Williams, S. L., Duncan, M. J., & Vandelanotte, C. (2016). Efficacy of interventions that use apps to improve diet, physical activity and sedentary behaviour: A systematic review. *International Journal of Behavioral Nutrition and Physical Activity*, 13(127) Retrieved from <https://doi.org/10.1186/s12966-016-0454-y>

schoolfreeware. (2017). Smartphone - tablet black and white free clipart icon. Retrieved from <https://openclipart.org/detail/276127/smartphone-tablet-black-and-white-free-clipart-icon>

Schüll, N. D. (2016). Data for life: Wearable technology and the design of self-care. *BioSocieties*, 11(3), 317-333. doi:10.1057/biosoc.2015.47

Speck, B., & Harrell, J. (2003). Maintaining regular physical activity in women: Evidence to date. *The Journal of Cardiovascular Nursing*, 18(4), 282-293. doi:10.1097/00005082-200309000-00007

Study focuses on strategies for achieving goals, resolutions. Retrieved from <https://www.dominican.edu/dominicannews/study-highlights-strategies-for-achieving-goals>

System usability scale. Retrieved from <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>

The Princeton Review. (2015). Students study the most. Retrieved from <https://www.princetonreview.com/college-rankings?rankings=students-study-most>

Thomas, N. (2015). How to use the system usability scale (SUS) to evaluate the usability of your website. Retrieved from <https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/>

Tofel, Kevin. (2015). Want your heart rate every 10 minutes from apple watch? don't move. Retrieved from <https://www.zdnet.com/article/apple-watch-heart-rate-every-10-minutes-motion/>

Tollmar, K., Bentley, F., & Viedma, C. (2012). Mobile health mashups: Making sense of multiple streams of wellbeing and contextual data for presentation on a mobile device. Paper presented at the 2012 6th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops,

Tremblay, M. S., Gray, C. E., Akinroye, K., Harrington, D. M., Katzmarzyk, P. T., Lambert, E. V., . . . Tomkinson, G. (2014). Physical activity of children: A global matrix of grades

comparing 15 countries. *Journal of Physical Activity & Health*, 11 Suppl 1(s1), S125.  
doi:10.1123/jpah.2014-0177

Under Armour, I. (2018). MyFitnessPal. Retrieved from <https://www.myfitnesspal.com/>

Under Armour, I. (2019). MapMyWalk. Retrieved from <https://www.mapmywalk.com/>

Vive precision electronic scale. (2019). Retrieved  
from <https://www.vivehealth.com/products/weight-scale>

Wade, J. (2017). Wearable technology statistics and trends 2018. Retrieved  
from <https://www.smartinsights.com/digital-marketing-strategy/wearables-statistics-2017/>

Welcome to the MyFitnessPal community! (2017). Retrieved  
from <https://myfitnesspal.desk.com/customer/en/portal/articles/2600288-welcome-to-the-myfitnesspal-community->

Why we should exercise - and why we don't. (2008). Retrieved  
from [https://www.health.harvard.edu/newsletter\\_article/why-we-should-exercise-and-why-we-dont](https://www.health.harvard.edu/newsletter_article/why-we-should-exercise-and-why-we-dont)

Withings. (2019). Withings health mate: A fitness, activity and health tracker app. Retrieved  
from <https://www.withings.com/be/en/health-mate>



# Appendices

## Appendix 1: Fitness App Survey

The following is the questions included in our initial survey, which we created before beginning app development. Not every respondent saw every question; some questions only appeared if a respondent answered a previous question in a particular way. Therefore, the questions are followed by a flow chart that shows every such relationship between questions.

As part of our background research for MQP, we invite you to take this survey about your health tracking habits. We are a group of seniors who plan to write an Android application that will allow the WPI community to track their health and fitness and compare their progress with others. The survey should take no longer than 5 minutes to complete. The information we gain from this survey will be used to help us determine what features we should build into our application to best fit the needs of the WPI community. The results of this study will be published in our final report, but it will contain no names, addresses, or any other identifiable information of individual responses. This survey is completely voluntary and respondents may skip any questions they prefer not to answer. If you have any questions about this study, please email [gr-openmhealthmqp@wpi.edu](mailto:gr-openmhealthmqp@wpi.edu).

---

How often do you use the following health and fitness apps/devices? (Note: iHealth is a company NOT related to Apple's Health app)

	Never	Once per month or less	Several times per month	Once per day	Multiple times per day/constantly
FitBit	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
GoogleFit	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
RunKeeper	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Misfit	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Withings	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
iHealth	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
MyFitnessPal	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

---

Do you use any fitness/health apps or devices not listed in the previous question?

- Yes
- No

---

Which other apps or devices do you use?

App/device 1	<input type="text"/>
App/device 2	<input type="text"/>
App/device 3	<input type="text"/>

---

How often do you use those other devices/apps?

	Once per month or less	Several times per month	Once per day	Multiple times per day/constantly
Samsung Health	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Apple Watch	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

---

Do you use any social features (e.g. MyFitnessPal Community, FitBit Community) built into your fitness/health apps, if any exist?

- Yes  
 No

---

Do you feel that using social features in these apps has helped you improve your health or fitness? If so, how have these features helped you?

- Yes   
 No

---

What do you use your device/app(s) to track?

- Step count  
 Sleep duration/cycles  
 Aerobic activity (i.e. walking, jogging, biking, etc.)  
 Weightlifting  
 Calories burned  
 Food intake  
 Body weight  
 Other

---

Are you interested in improving your fitness or health in any of these areas? If so, which ones?

- Step count
- Sleep duration/cycles
- Aerobic activity (i.e. walking, jogging, biking, etc.)
- Weightlifting
- Calories burned
- Food intake
- Body weight
- Other
- None of the above

---

How interested would you be in using an application that compares fitness and health data across the WPI community? Your specific data would be private only to you and those with whom you choose to share it.

- Extremely interested
- Moderately interested
- Slightly interested
- Not interested at all

---

In the application described in the previous question, which types of data would you like to be able to compare to those of your friends?

- Step count
- Sleep duration/cycles
- Aerobic activity (i.e. walking, jogging, biking, etc.)
- Weightlifting
- Calories burned
- Food intake
- Body weight
- Other

---

Are there any additional features you'd like to see in this application?

- Yes
- No

---

I am a...

- WPI student
- WPI faculty member
- Other

---

What is your gender?

- Male
- Female
- Prefer not to say
- Other

---

What is your age?

---

How would describe your activity level?

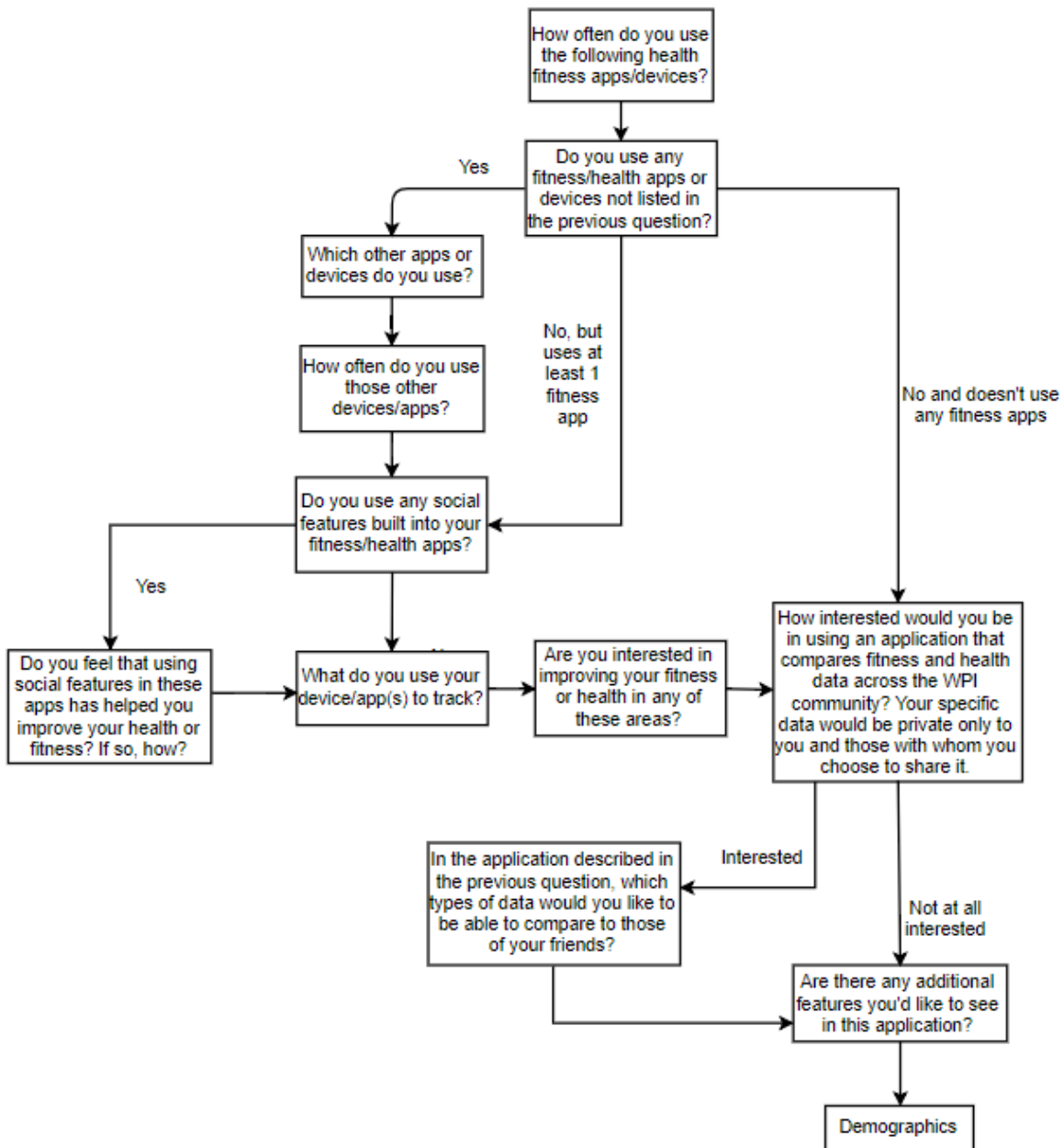
- Very active
- Active
- Lightly active
- Not active

---

What kind of mobile phone do you have?

- iPhone
- Android
- Other

Since some of the survey questions depended on answers to previous questions, the survey utilized decision logic to display the appropriate questions to the participants. This flow chart represents the logic used by the survey.



## Appendix 2: Evaluation Survey

The following is the survey we provided to beta testers during the evaluation of HerdHealth. After using the app for a short amount of time, participants were asked to answer these questions. Unlike in the initial survey, all respondents were shown every question.

As part of our evaluation efforts for MQP, we invite you to take this survey about your experience using our app. The survey should take no longer than 5 minutes to complete. The information we gain from this survey will be used to help us determine what people like and dislike about our app, as well as how easy it is to use. The results of this study will be published in our final report, but it will contain no names, addresses, or any other identifiable information of individual responses. This survey is completely voluntary and respondents may skip any questions they prefer not to answer. If you have any questions about this study, please email [gr-openmhealthmqp@wpi.edu](mailto:gr-openmhealthmqp@wpi.edu).

---

Please rate your level of agreement with the following statements

	Strongly agree	Somewhat agree	Neither agree nor disagree	Somewhat disagree	Strongly disagree
I think that I would like to use the system frequently	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I found the system unnecessarily complex	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I thought the system was easy to use	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I think that I would need the support of a technical person to be able to use this system	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I found the various functions in this system were well integrated	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I thought there was too much inconsistency in this system	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I would imagine that most people would learn to use this system very quickly	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I found the system very cumbersome to use	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I felt very confident using the system	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
I needed to learn a lot of things before I could get going with this system	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

What feature(s) did you like the best? Why?

What feature(s) did you least like, or would be least likely to use? Why?

Are there any types of data (e.g. floors climbed, carbohydrates, etc.) that are not currently shown in the app which you would like to see?

Do you have any suggestions regarding the app's appearance?

What features would you like to see in the future?

I am a...

- WPI student
- WPI faculty member
- Other

What is your gender?

- Male
- Female
- Prefer not to say
- Other

What is your age?

How would you describe your activity level?

- Very active
- Active
- Lightly active
- Not active

What kind of mobile phone do you have?

- iPhone
- Android
- Other

### Appendix 3: System Usability Scale Results and Scoring Details

The System Usability Scale (SUS) is a standardized method of measuring the usability of a system (System Usability Scale, *n.d.*). The responses on the SUS range from “Strongly Disagree” (1) to “Strongly Agree” (5). All of the odd statements have negative connotations, while the even statements are more positive. Therefore, to give the system a perfect score, the respondent would respond “Strongly Disagree” on all the odd statements and “Strongly Agree” on all the even statements (Thomas, 2015).

The SUS supplies a method for scoring the results in order to quantify a system’s usability. To score one response, the researchers must subtract 1 from all odd numbered responses, then sum the results. Next, the researcher subtracts the even responses from 5 and sums those results. Adding the odd-numbered sum to the even-numbered sum will result in a score out of 40 possible points. In order to standardize this score out of 100 and get the final score, multiply by 2.5. Note that this is not a percentage and should not be interpreted as such (Thomas, 2015). The scoring guidelines and our results can be found below.

#### Scoring Guidelines

Score $\geq$ 80.3	A
$63 \leq$ Score $\leq$ 80.3	C
Score $\leq$ 51	F

#### Our Results

Questions	Responses																						
1	3	2	4	2	4	3	4	2	5	5	4	4	4	5	5	5	3	4	5	1	3	5	5
2	3	1	2	1	1	2	1	1	1	1	2	2	1	2	1	1	2	1	1	1	2	2	2
3	4	5	5	5	4	4	5	4	4	5	4	4	5	5	5	5	4	4	4	4	4	4	5
4	1	1	1	1	1	1	1	1	1	1	1	1	1	5	1	1	1	1	1	1	1	2	1
5	4	5	4	4	3	4	5	4	5	5	4	5	5	5	5	5	4	2	5	3	4	5	1
6	3	1	2	1	2	1	2	3	1	1	2	1	2	2	1	1	3	2	1	1	3	2	1
7	4	4	4	5	4	5	5	4	4	4	5	5	5	4	5	5	4	4	4	4	5	5	5
8	2	2	2	1	1	2	1	2	2	1	3	1	2	2	1	1	3	2	1	1	4	2	1
9	3	4	4	4	4	4	5	4	4	5	5	5	5	4	5	4	3	4	4	3	4	4	4
10	2	2	1	1	1	2	1	1	4	1	1	1	2	2	1	2	1	3	2	1	2	2	4

Individual Scores	27	33	33	35	33	32	38	30	33	39	33	37	36	30	40	38	28	29	36	30	28	33	31
-------------------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

<b>Mean</b>
33.13043 /40

<b>Standardized Score</b>
82.82609 /100