

Health and Fitness Smart Device

Group 6

Prepared By: Amber Haynes, Tiyon King, Jenna Krause, Mya Odrick,
Devvrat Patel, Andrew Rezk, Maria Rios, Shivani Sunil, Hedaya Walter
<https://sites.google.com/scarletmail.rutgers.edu/betterfitnesscom/home?authuser=1>

February 23, 2020

Software Engineering

Table Of Contents

Project Ownership	2
Individual Breakdown	2
Customer Problem Statement	4
Glossary of Terms	4
Proposal	5
Problem Statement	6
User Stories	9
Functional Requirements Specification	11
Use Cases	12
Use Case Diagram	15
Traceability Matrix	16
Fully-Dressed Description	17
System Sequence Diagrams	21
User Interface Specification	23
Preliminary Design & User Effort Estimation	23
Domain Analysis	26
Domain Model	26
Concept Definitions	27
System Operation Contracts	32
Project Size Estimation (Use Case Points)	34
Project Management	39
References	40

Project Ownership

Calendar Sync & Workout Reminders

- Tiyon King & Jenna Krause

Calorie Tracker

- Mya Odrick, Maria Rios, & Hedaya Walter

Music App Integration

- Amber Haynes & Shivani Sunil

Safety & SOS Feature

- Devvrat Patel & Andrew Rezk

Individual Breakdown

			Amber Haynes	Tiyon King	Jenna Krause	Mya Odrick	Devvrat Patel	Andrew Rezk	Maria Rios	Shivani Sunil	Hedaya Walter
Project management		10									
Customer Problem Statement	Prob. statm.	5		20%	20%			20%	20%	20%	
	Glos sary	4				100%					
System Requirements	Funct. req'ts	2		25%				25%		25%	25%
	Nonfct. req'ts FURPS+	2		25%				25%		25%	25%
	UI req'ts	2		25%				25%		25%	25%
Functional Requirements Specification	Stkhld. Actors Goals	2		14%	14%	14%		14%	14%	14%	14%
	Use c. casual descr.	8	11%	11%	11%	11%	11%	11%	11%	11%	11%
	Use case diag	5			100%						

	Use c full descr.	10	25%		25%	25%	25%				
	Sys. seq. diag	5	25%		25%			25%	25%		
User Interface Specs	Prelm design	4				25%	25%		25%	25%	
	Effort estim	11				50%			50%		
Domain Analysis	Conc epts	10	25%		25%	25%	25%				
	Assoc	5	25%		25%	25%	25%				
	Attrib	5	33%		33%	33%					
	Cont racts	5	50%						50%		
Project Estimation			33%	33%					33%		
Plan of Work		5		50%		50%					
Ref's		5	33%	33%					33%		

Customer Problem Statement

Glossary of Terms

Alert - An automated message containing the user's location and heartbeat. An alert can be sent to the user's most important contacts in the event that the user feels unsafe.

Calorie Deficit - A calorie deficit occurs when the amount of calories consumed is less than the amount needed to maintain current body weight. If the user's goal is to lose weight, we will recommend a caloric deficit diet and workout. We will suggest low-calorie meals and calorie-burning exercises.

Calorie Surplus - A calorie surplus occurs when the amount of calories consumed is greater than the amount needed to maintain current body weight. If the user's goal is to gain weight or muscle, we will recommend a caloric surplus diet and workout. This means we will recommend healthy high-calorie food items and muscle gaining exercises.

Favorites List - The user can add specific meals and songs to their Favorites List, which allows them to access those items quicker. Upon clicking on one of their favorite items, they will receive information like the number of calories in the meal, or the best exercises to perform while listening to their songs.

Freetime - Freetime is defined as a block of time where the user doesn't have anything planned. The user can manually enter their free time and we will suggest the user to exercise during that time.

Reminder - The user will receive notifications as reminders to suggest them to stick to their scheduled workout, input their daily calorie intake, and incorporate certain food or exercises to meet their goals. However, these reminders can be manually turned off.

Tempo - The tempo of a song refers to the speed or pace of the music. We will use the user's desired workout and recommend music based on the tempo, as well as the user's preferred genre of music.

Proposal

We will be basing the general platform from that of group number five in the 2014 Fall Semester. The previous group focused on using the Fitbit to educate the users, and help them maintain a healthy lifestyle; however, we want to motivate users to continue to use their devices when they lose their drive to workout or eat healthily.

The user will be able to sync their Google calendar to the watch. This would allow the user to identify specific workout times. This feature, though optional, is best used for those who input their daily schedules into their calendar. The purpose of the feature is to better motivate the user to implement workouts on a greater basis. Using this feature, the device will seek out the free time, the time when no event is planned, during the day that could be used for exercising. The user will be able to select a range of time in which the feature may look for free time, such as 8 AM to 10 PM. We will be implementing a reminder system to motivate the person wearing the device to stick to their already scheduled workouts, as well as use their free time to exercise on the days where no workouts have been already scheduled. These reminders are simply to workout, whether that be at home or at a nearby fitness center; the feature does not specify. When these reminders occur, the user will have an option to snooze it - like an alarm - for a later reminder, check-in and say that they are working out, indicate that they are at an event that was not inputted, or indicate that they do not wish to workout that day.

Another added feature will be an improvement of the calorie tracker. To motivate the user to continue their healthy lifestyle, the new calorie tracker will send the user food and workout recommendations based on their calorie intake and outtake ratio. Depending on their weight goals, they will be proposed ways to either burn or consume more calories. To motivate the user to use this function, we will send out a reminder for them to input their daily calorie intake and send notifications regarding if they met their goal for the day.

Another added feature will be a virtual workout music playlist curator for users. This feature will take information from users involving their preferred types of workouts and times which they like to workout, and curate playlists for them for different workout days based on the beats per minute (bpm) of the songs. In contrast to other features currently on the market, this part of the site will closely relate the type of exercise (and intended time spent doing the exercise) to decide

which and how many songs to add to the playlist. For example, if a user says that they want to jog for 15 minutes then walk for 20, a playlist will be created with 15 minutes of fast-paced music (from their preferred genres) and 20 minutes of slower paced music that they can play when they workout. This also will keep the user on track when they are working out. This also opens the opportunity for our feature which is connected to this, a feature that rewards users with online subscriptions and other prizes based on if they complete their workout.

Problem Statement

As a consumer, it becomes very difficult to find time throughout the day to work out. Whether working or in school, incorporating workouts into an already busy schedule becomes stressful and discouraging. In addition to finding time, it also becomes hard to stay motivated to stick to exercising during the time I've devoted to it. Using the calendar and reminder features on the device, the overwhelming and discouraging aspect of finding the times to dedicate to exercise are taken out of the equation. This relief combined with the additional feature of motivational reminders for the time I had already dedicated to working out take away major factors that previously prevented me from sticking to the plan.

For someone who uses their Google calendar to keep track of events and their schedule, being able to easily incorporate my already created calendar into a new device makes the experience seamless and much less stressful. The only factors that I will have to worry about are selecting the range of time in which the reminder feature will look for free time day-to-day and the amount of time needed in order to exercise, making the set-up experience quick and painless. Picking the amount of free time needed in order to exercise will give me the flexibility for whether I work out at home or if I go to the gym to work out. Once the range of times and amount of time needed to exercise are set, I will receive motivational reminders or notifications informing me of when I have enough free time to incorporate a workout into my schedule.

These features will allow me to be better motivated in order to implement workouts into my schedule on a greater basis. As most know, constant notifications can become quite annoying, usually prompting me to turn the reminders off completely. When the reminders occur, I will have the option to snooze it - like an alarm - for a later reminder, check-in and say that I am working out, indicate that I am at an event that was not inputted, or indicate that I do not wish to workout that day. The ability to snooze or dismiss the reminders and

notifications with a wide range of options will lower the chance that I will disable the feature completely.

As a user, I need a way to keep track of my intake of calories and keep track of how many calories I've burned. I want to be able to see my progress and check how much I should be eating based on the exercises I complete. I can input my details into the app at the beginning of my fitness journey such as my current weight, height, age. I can let the app know my ultimate goal and the app can suggest different foods I can eat based on other users that have similar details and have had success. The calorie tracker feature will incorporate a search bar to make it easier for me to search up the foods that I am eating and will give me the number of calories each food contains. The calorie tracker feature can then use this information to calculate how many total calories I've consumed.

In addition, it will be difficult for me to know which exercises are better suited for me and how effective it will be. I will rely on the app to give me different exercise suggestions based on those that have worked for other users with similar profiles. I can choose which ones to follow and put that information into the app. For example, if I want to gain muscle, I will receive food recommendations that will help me achieve this. Likewise, if I want to lose fat, I will be given the appropriate recommendations. These recommendations will save me time and make it easier for me to make the right choices.

Because I have a busy schedule and can be forgetful at times, I will depend on the app to send me a reminder at the start of the day to let me know what type of exercises I should complete. I can adjust the frequency of the reminders so I could choose to have the app remind me at the meal times that I input into the system or just have it remind me at the start of the day. At the end of each day, the app will let me know the progress I've made that day. I will receive a notification of what I've burned off and what I have consumed. Seeing the progress I've made will motivate me to keep working hard towards my goals

We will work on a feature that is necessary for those meaning to exercise with the watch on. For those schedules that permit running late at night or early morning, it's important they feel safe. So this pair will send a reminder to the user that this is an option and make sure that when it is implemented that it triggers a set of commands in the following order: Obtain all possible information such as time, place, etc. Then notify the select list the user creates. This will be a safety feature

that will give the user an easy option to get help when needed during an emergency.

This feature helps inform the user's closest friends and family about any alarming concerns if the user deems it necessary. This is ultimately to give the user a sense of safety. Some of the emergency features will include emergency calling (if the watch supports it) and notifying emergency list if the user is in danger but, unable to call themselves. This feature is really important as we want to make sure that everyone who uses our app can feel safer. A lot of elderly people would also find this particularly helpful as there are many cases where they'd be in trouble and found it really hard to reach someone. This feature can also be used in a dangerous situation when people don't have access to their phones and clicking a button on their Fitbit is much easier to access. This feature will not only share that the user is in danger but also share their live location to the select list of people. One other feature we would have is that alerting the close friends list even when the user cannot. For example, the app will know if the user is having a health issue based on their breathing pattern. This way if the user is having an emergency then the watch will automatically alarm their close friends and also call for help if that is selected. The app will also have a record of the surroundings and the user's running location so if there is anything wrong going outside.

User Stories

Table 1-1: Calorie Tracker Requirements

ID	Priority Weight	Requirement
REQ-1	3	As an authorized user, I will be able to take a picture of my food to input calorie intake.
REQ-2	2	As an authorized user, I will be able to enter my fitness goals.
REQ-3	5	The systems will suggest meals and exercises based on the goals of the user.
REQ-4	7	The system will use the user's heart rate to calculate the number of calories burned during a workout.
REQ-5	3	The system will display the number of calories burned and the number of calories consumed.
REQ-6	7	As an authorized user, I will be able to search for my meals through a search bar to input my calorie intake.
REQ-7	3	As an authorized user, I will be able to save frequent or favorite meals/snacks for a faster and easier experience.

Table 1-2: Calendar Requirements

ID	Priority Weight	Requirement
REQ-8	6	As an authorized user, I will be able to sync my Google Calendar with the application.
REQ-9	3	As an authorized user, I will be able to snooze workout reminders for a later time.
REQ-10	5	As an authorized user, I will be able to toggle workout reminders on or off.
REQ-11	3	As an authorized user, I will be able to 'check-in' with the app to inform that I am sticking to my scheduled workouts.
REQ-12	9	The application will suggest times the user should workout, based on user's free time.

Table 1-3: Music Curator Requirements

ID	Priority Weight	Requirement
REQ-13	9	As an authorized user, I will be able to pick a type of workout I would like to do and the app should provide me with playlists with BPMs suited for that workout.
REQ-14	8	My playlist options should also incorporate the artists I prefer
REQ-15	9	I should be able to set a workout goal in terms of calories or steps and upon achievement of those goals, I should be able to activate a free membership reward.
REQ-16	3	I should be able to save the playlists I prefer and shuffle the same set next time.
REQ-17	4	I should be able to rely on the app to generate new playlists for me, as well
REQ-18	2	The music tempo should be correlated to my heartbeat while working out

Table 1-4: Safety & SOS Requirements

ID	Priority Weight	Requirement
REQ-19	8	As an authorized user, my watch will alert friends and family upon a press of a button
REQ-20	3	The watch should detect whether the user is running or user sets up exercise thus activating a safe run(feature with high sensitivity to any movements that alerts people)
REQ-21	5	I should be able to go out on a run with my watch knowing that especially if it's at night
REQ-22	7	The watch can detect weather and precipitation and alarm me if it's not safe or if there's a state of emergency
REQ-23	6	The watch should be able to detect user's health issues and warn their family and friends
REQ-24	5	The watch should not have a false alarm and create a sense of urgency.

Functional Requirements Specification

Stakeholders

Identify anyone and everyone who has an interest in this system (users, managers, sponsors, etc.). Stakeholders should be humans or human organizations.

- Users
- Nutritionists
- Music Services
- Google Calendar
- Emergency Responder

Actors and Goals

Identify the *roles* of people or devices that will directly interact with the system, their *types* (initiating vs. participating) and the *goals* of the initiating actors.

- Users
 - Role: Implement this technology into their lives
 - Type: Initiating
 - Goal: To maintain or achieve their fitness goals
- Google Calendar
 - Role: Provide access to previously created Google calendars
 - Type: Participating
- Nutritionists
 - Role: Gives input on what foods and exercises to recommend to users.
 - Type: Participating
- Music Services
 - Role: Provide access to music streaming services
 - Type: Participating
- Emergency Responder
 - Role: Upon request, offer assistance for users who feel unsafe
 - Type: Participating

Use Cases

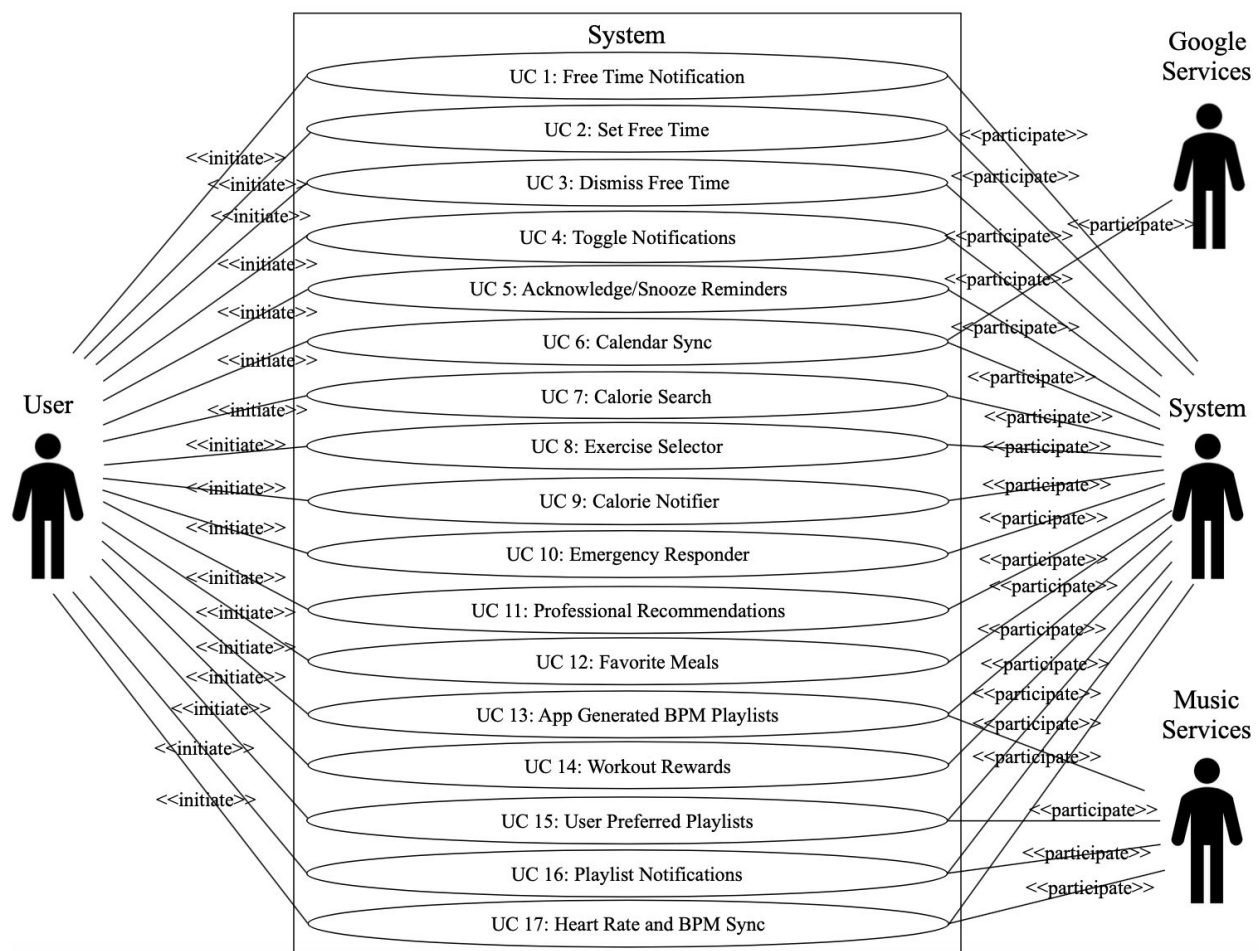
Actor	Actor's Goal	Functional Requirements	Use-Case Name
System	Remind users during the	Provides users with	Free Time

	pre-set period of the day if they have enough free time to exercise.	time throughout the day that they can implement a workout.	Notifications (UC#1)
User	Users will be able to set the minimal amount of free time needed to exercise as well as the time interval in which free time reminders are enabled.	Allows the user to customize the free time interval and amount of time required for a work out.	Set Free Time (UC#2)
System	If the user indicated that they exercised or if they dismissed the free time notification free time notifications will be disabled until the start period of the following day.	Turn off free time or work out notifications until the free time interval of the next day.	Dismiss Reminders (UC#3)
User	Users will be given the option to enable or disable free time notifications.	Toggle free time notifications ON/OFF.	Toggle Notifications (UC#4)
User	Users will be able to acknowledge, snooze, or accept the work out reminders.	Encourages the user to stick to their goal of exercising.	Acknowledge/Snooze Reminders (UC#5)
System	Remind the user to exercise, again, after a given period of time.	Waits 30 minutes before checking if the amount of free time is enough to implement a work out.	
User	Users will be able to sync their Google Calendars to the calendar on their device.	Allows users to seamlessly add their already created calendars to their device.	Calendar Sync (UC#6)
User	Users will be able to use a search bar to look up the calories in their food.	Lets users easily add the calories eaten into the system.	Calorie Search (UC #7)

User	Users will select what exercises they have completed for the day.	System will receive that information and know how many calories were burned based on pre-assigned values.	Exercise Selector (UC #8)
User	Users will be able to select a future workout based on the number of calories they will potentially burn	System will determine the number of calories burned based on the user's past exercise experiences	
System	Keeps track of the calories consumed and burned.	Notifies users of the calories lost or gained at the end of each day.	Calorie Notifier (UC #9)
User	Users will be able to notify a list of people if they are in danger.	The system will have a list of people made by the user, and the system will notify those people if the user presses a button	Emergency Responder (UC #10)
System	The system will automatically notify the close family list if the user is in a medical emergency.	The system will be able to identify user's health emergencies based on their heartbeat and notify the select list of people if need be.	
User	Users will be able to interact with recommendations suggested by nutritionists	The system will have a database of healthy recommendations suggested by professionals.	Professional Recommendations (UC #11)
User	Users will be able to save frequent/favorite meals for a quicker and easier meal-tracking experience.	App will display the list of favorite meals under the meal search bar.	Favorite Meals (UC #12)
User	Users will be able to select a certain type of workout and the app will generate	App will display the selected playlists and the user will have the	App generated BPM playlists (UC #13)

	BPM based playlists based on the intensity of the workout	option to save them for future workouts	
System and User	On the 1st of every month, the user will be prompted to enter a workout goal, either in terms of calories or steps.	The app will then reward the user, in terms of subscriptions of one month	Workout rewards (UC #14)
User	Users will be able to integrate spotify/apple music from their phone	The app will also generate playlists based on the artists the user likes/prefers and will allow the user to save them	User preferred playlists (UC #15)
System	The User will receive daily revised playlists from the app, to keep them motivated as listening to same music overtime gets monotonous	The app will generate new playlist notifications for the user, and perhaps in conjunction with the workout reminders to increase the motivation factor.	Playlist notifications (UC #16)
System	The user will experience a change in music tempo in conjunction with their heart rate.	The app will change songs/ music tempo by detecting the user's heart rate, high BPM songs will play	Heart rate and BPM sync (UC #17)

Use Case Diagram



Traceability Matrix

Req.	PW	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC 10	UC 11	UC 12	UC 13	UC 14	UC 15	UC 16	UC 17
1	3							x						x				
2	2								x						x			
3	5							x	x			x						
4	7								x	x							x	
5	3							x		x								
6	7							x										
7	3												x					
8	6						x									x		
9	3					x												
10	5			x	x													x
11	3			x		x												
12	9	x	x															
13	9													x		x	x	
14	8													x			x	
15	8														x			
16	3															x	x	
17	4													x		x		
18	2																x	x
19	8										x							
20	3										x							
21	5					x	x				x							
22	7										x							
23	6										x							
24	5																	
Max PW		9	9	5	5	5	6	7	7	7	8	5	3	9	8	9	8	5
Total PW		9	9	8	5	11	11	13	13	10	15	5	3	15	10	15	15	7

Fully-Dressed Description

User Case UC- 5 : Acknowledge/Snooze Reminders	
Related Requirements:	REQ9 and REQ11 as stated in Table 1-2
Initiating Actor:	User
Actor's Goal:	To indicate that they have/are exercising or to snooze the notification for a later reminder
Participating Actors:	System
Preconditions:	<ul style="list-style-type: none"> • User has entered minimal amount of free time needed to exercise • User has inputted the time interval in which free time reminders are enabled • Current time is within users previously set notification interval • Enough free time to exercise
Postconditions:	<ul style="list-style-type: none"> • Reminders will be stopped until the next free time interval of the following day (Acknowledge) • Another reminder will occur after 30 minutes if enough time to exercise is still available (Snooze)
Flow of Events for Main Success Scenario:	
→	1. User has inputted the time interval in which free time reminders are enabled
→	2. User has entered the minimum amount of time needed to exercise
←	3. System scans calendar to see if there is enough free time to exercise
→	4. User snoozes/acknowledges the reminder
←	5. System disables free time notifications until the next day/looks for free time again in 30 mins

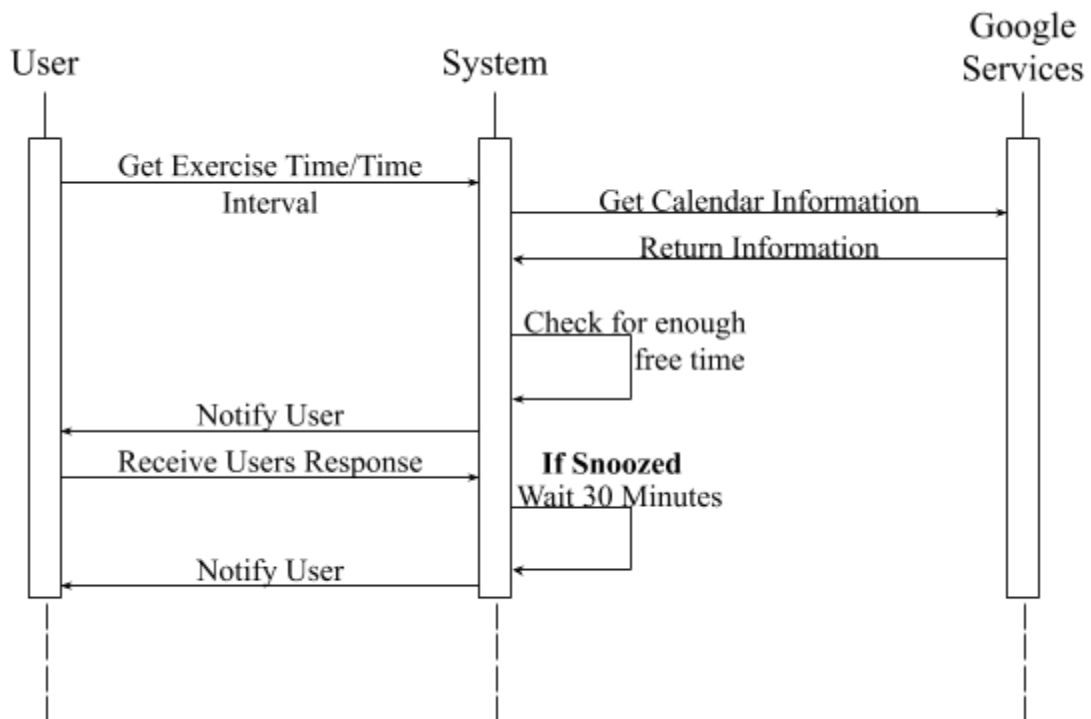
User Case UC - 7: Calorie Search	
Related Requirements:	REQ1, REQ3, REQ5, and REQ6 as stated in Table 1-1
Initiating Actor:	User
Actor's Goal:	To use a search bar to look up the number of calories in food.
Participating Actors:	System
Preconditions:	<ul style="list-style-type: none"> • User hasn't entered daily calorie intake • User will like to know the number of calories in certain meals
Postconditions:	<ul style="list-style-type: none"> • The number of calories will be added to the user's calorie tracker. • User will be redirected back to the search bar page.
Flow of Events for Main Success Scenario:	
→	1. User enters a food item name
←	2. System (a) provide calorie count for average portion size (b) prompts user to add calories to calorie tracker
→	3. User (a) edits the portion size and/or (b) confirms calorie count
←	4. System updates the user's daily calorie count

User Case UC- 13: App-Generated BPM Playlists	
Related Requirements:	REQ-13, REQ-14, REQ-16, and REQ-18, as stated in Table 1-3
Initiating Actor:	User
Actor's Goal:	To receive personalized workout playlists based on inputted intended exercises.
Participating Actors:	System
Preconditions:	<ul style="list-style-type: none"> • User enters intended workout and exercises, along with duration, into the playlist section of the app. • User enters preferred genres for listening. • User gives the app permission to use personal music subscription account to create a playlist.
Postconditions:	<ul style="list-style-type: none"> • The system creates a playlist based on intrinsic data that matches songs to intended workout, workout time, and genre preference and automatically saves workout and playlist. • The system notifies the user when the playlist is ready to use. • The system connects to a music app to play curated songs from a playlist in correct order when prompted.
Flow of Events for Main Success Scenario:	
→	1. User enters intended exercises and preferred genres into the system.
→	2. User gives system access to their 3rd party music account.
←	3. System creates playlist using data inputted by the user and saves the playlist and workout data for future playlists.
←	4. System notifies user of playlist creation.
→	5. User prompts the system to play a playlist when desired.
←	6. System plays a playlist using the music app when prompted.

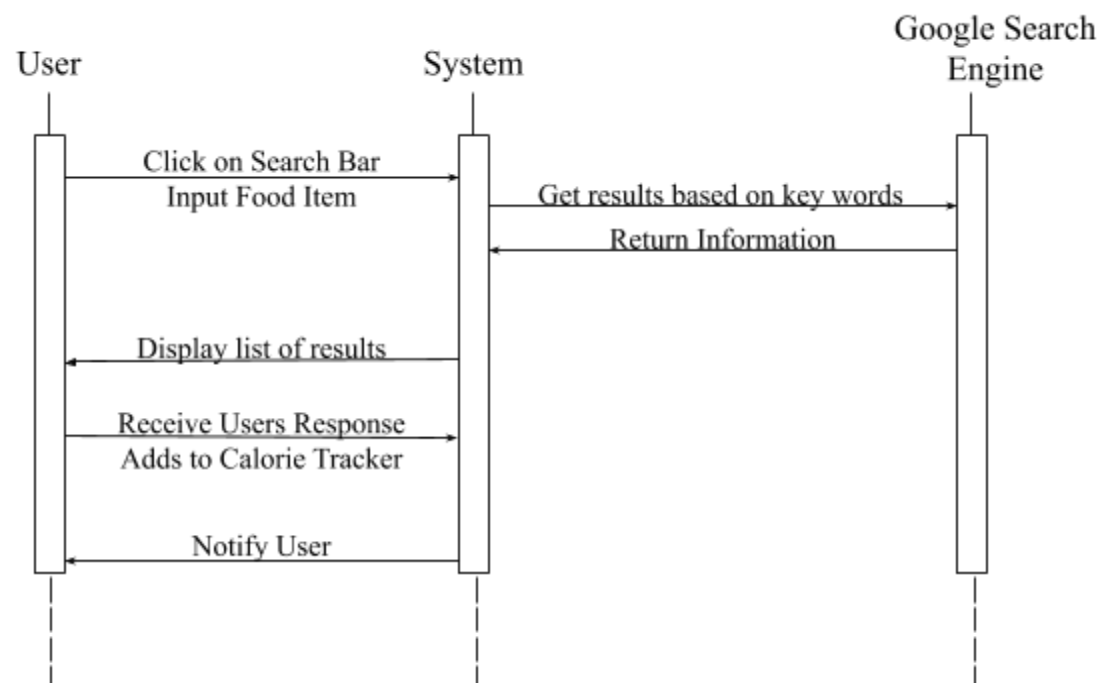
User Case UC- 10 : Emergency Responder	
Related Requirements:	REQ19, REQ20, REQ22 and REQ23 as stated in Table 1-2
Initiating Actor:	User
Actor's Goal:	Users will be able to notify a list of people if they are in danger.
Participating Actors:	System
Preconditions:	<ul style="list-style-type: none"> • User must have a premade emergency contact list that they have set up • User needs the device with them • User would need to be in running state or user manually triggers the command via a button • Location must be turned on, and data must be stored • There must be a signal for the phone to transmit information
Postconditions:	<ul style="list-style-type: none"> • Users device information is sent to the people on the list • Users heartbeat is then monitored and giving a refreshed update
Flow of Events for Main Success Scenario:	
→	1. User feels unsafe and triggers the button
←	2. System collects location information, heartbeat, and all other necessary and vital data
←	3. System then packages this data and send them to the provided list of contacts that user had setup
←	4. System then sends data, and has an auto update depending on the preference of the user.

System Sequence Diagrams

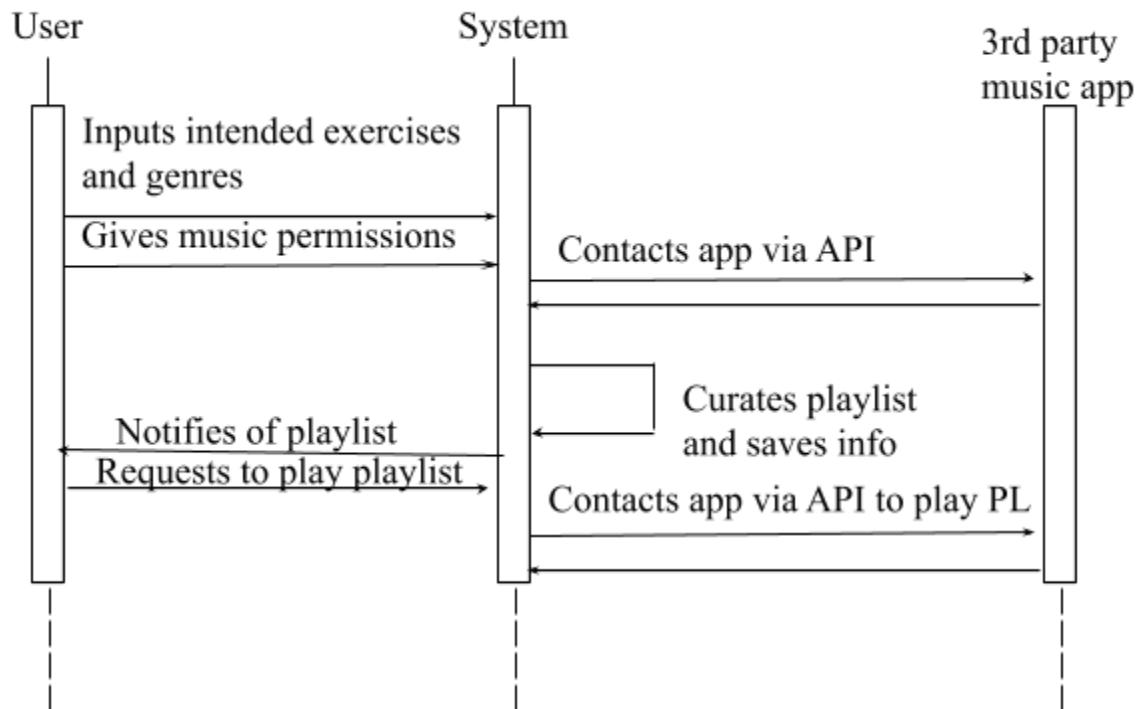
System Sequence Diagram for Use Case #5



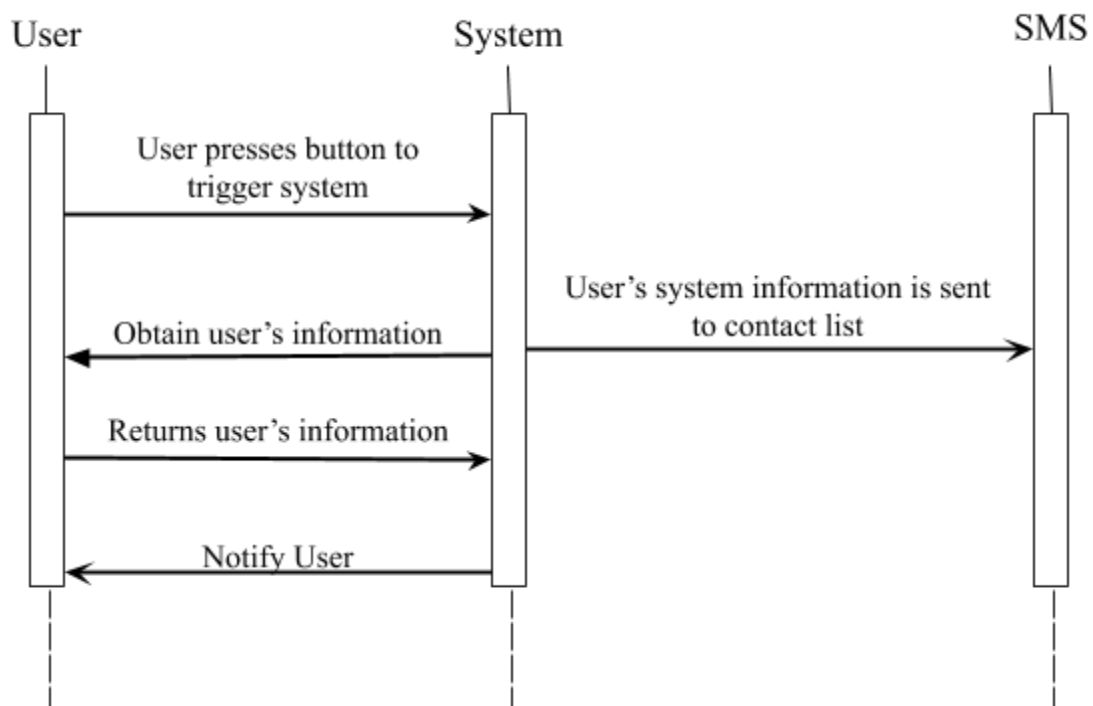
System Sequence Diagram for Use Case #7



System Sequence Diagram for Use Case #13



System Sequence Diagram for Use Case #10



User Interface Specification

Preliminary Design & User Effort Estimation

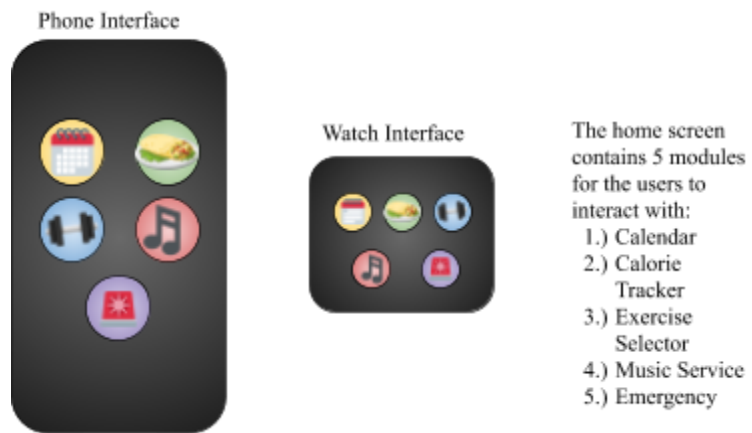


Figure 1.1 - Homescreen

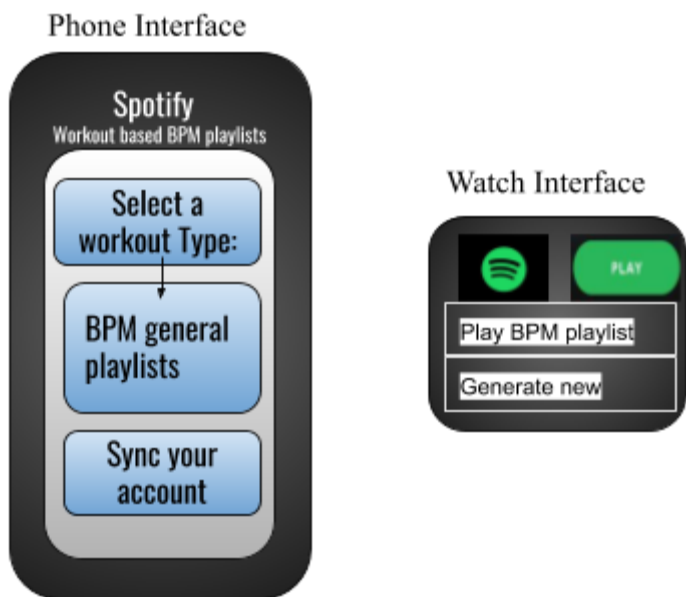


Figure 1.2 - Music Curator

User Effort Estimation:

For the interface of the calorie tracker, the user will see a goal meter, three options to enter their calorie count, a favorite button and a back button. The goal meter will be calculated with the user's information about their calorie intake/outtake ratio. The 'Search for exercises' button will allow the user to browse through the recommended exercises, as well as select their current workouts. The 'Search for meals' button will allow the user to search for meals and receive the calorie count for them. The 'Input Calories' button will allow the user to manually input their calories. The 'favorites' button will allow the user to access the user's favorite exercises and meals. The 'Go back' button will bring the user back to the homescreen.

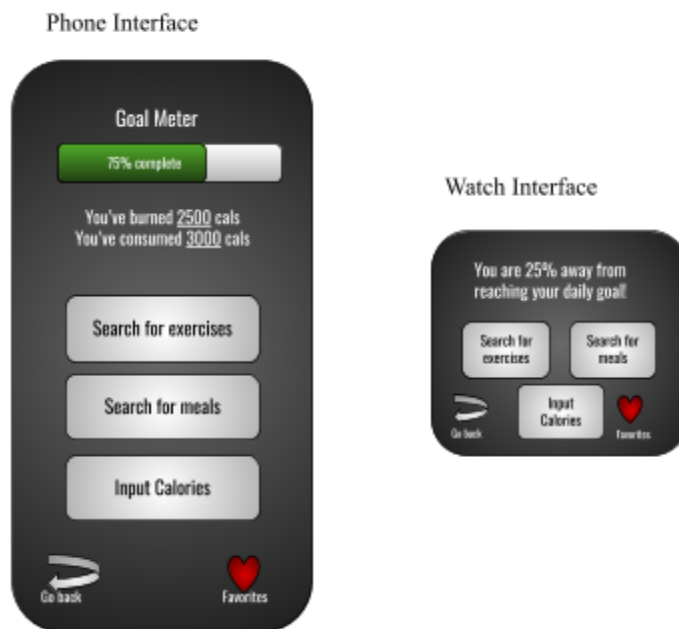


Figure 1.3 - Calorie Tracker

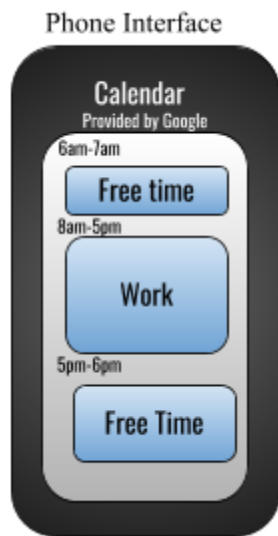


Figure 1.4 - Calendar Interface



Figure 1.5 - Exercise Selector

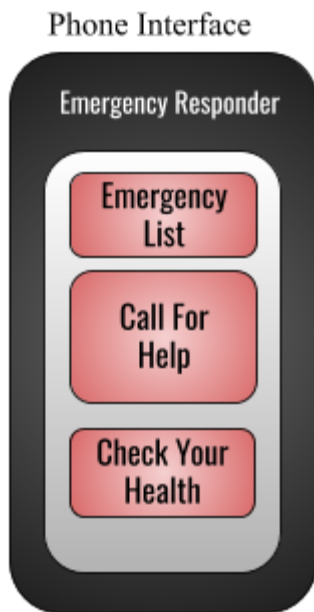


Figure 1.6 - Emergency Responder

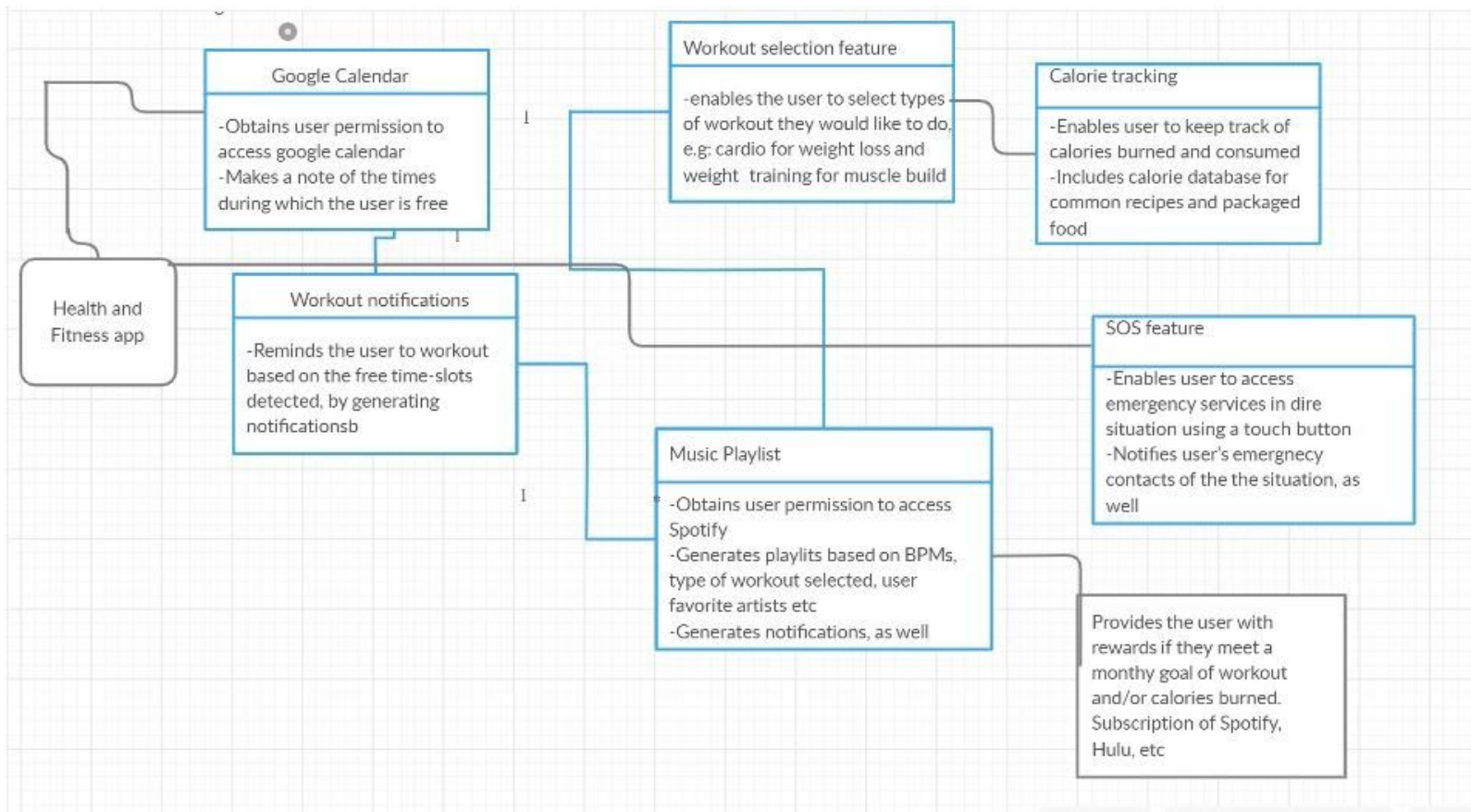


User Effort Estimation:

The interface for our application is simple with a few buttons and clear titles per page. The user will be able to access any page by clicking the icon on the home page and can return by clicking on a back button. Each page follows a similar setup to simplify the experience for the user. Overall, the user will not require much effort to use our application.

Domain Analysis

Domain Model



Concept Definitions

Responsibility	Concept
R1: Set the users free time interval and workout time	Calendar
R2: Calculate and keep track of users free time	Calendar
R3: Send notifications to work out during enough free time	Calendar
R4: Snooze/Dismiss/Acknowledge free time notifications	Calendar
R5: Enable/Disable free time reminders	Calendar
R6: Gather data from Google Calendar	Calendar
R7: Calculate user's calorie intake/outtake ratio	Calorie Tracker
R8: Display the amount of calories consumed/burned	Calorie Tracker
R9: Recommend exercises/ meals to meet goal	Calorie Tracker
R10: Compile list of most frequent exercises/meals	Calorie Tracker
R11: Display list of common exercises for selection and additional options for each exercise post-selection	Workout Selector
R12: Compiles and saves workouts under user-chosen name	Workout Selector
R13: Prompts user for 3rd-party music subscription login and permissions	Music Playlist
R14: Accesses user-intended workout when user prompts creation of playlist	Music Playlist
R15: Matches songs to each exercise intensity/BPM and length and compiles into playlist	Music Playlist
R16: Saves playlist under unique name for later use	Music Playlist
R17: Allows the user to create a list of close contacts for emergency	SOS Feature
R18: Allows the user to notify the close contacts list in case of emergency	SOS Feature
R19: Checks the status of the user's health	SOS Feature

Association Definitions

Concept Pair	Association Description	Association Name
Calendar ↔ SyncCalendar	Calendar calls SyncCalendar to sync the users existing Google Calendar with the calendar on the device.	Get Request Send Response
Calendar ↔ SetFreeTime	Calendar calls SetFreeTime when the user is initially setting the free time interval and workout time needed and when the user is updating those set times.	Set Request
Calendar ↔ CalcFreeTime	Calendar calls CalcFreeTime to check if there is enough free time, within the free time interval, to exercise.	Get Request Send Response
Calendar ↔ ToggleFreeTime	Calendar calls ToggleFreeTime to enable and disable free time notifications.	Set Request
Calendar ↔ FreeTimeResp	Calendar calls FreeTimeResp to store the users response to a free time notification.	Get Request Send Response
Calendar ↔ NotifyFreeTime	Calendar sends a notification to the user when there is enough free time, within the free time interval, to exercise.	Send Response
Calorie Tracker ↔ CalRatio	Calorie Tracker invokes CalRatio to determine the ratio of calorie intake to outtake	Send Result
Calorie Tracker ↔ FoodRec	FoodRec will use the Calorie Tracker, along with user preferences, to recommend meals that will help the user meet their goals	Get Request Send Response
Calorie Tracker ↔ ExerciseRec	ExerciseRec will use the Calorie Tracker to recommend exercises that will help the user burn enough calories to meet their goals	Get Request Send Response
Calorie Tracker ↔ FreqList	FreqList will compile a list of the user's most frequent exercises and meals	Get Request Send Response

CalDisp ↔ CalRatio	CalDisp will use the information from CalRatio to display the User's calorie consumption and calories burned	Send Result
Workout Selector ↔ ExerSelect	ExerSelect will display a list of exercises for user selection, and expand to provide options for time or amount when exercise is selected.	Get Request Send Response
Workout Selector ↔ ExerSave	ExerSave will take user-inputted name and save the workout under that name in the user's account.	Get Request Send Response
Music Playlist ↔ MusicSync	MusicSync will prompt the user to input their 3rd party music subscription info and give app permissions.	Get Request Set Request Send Result
Music Playlist ↔ WorkChoose ↔ ExerSave	WorkChoose will display a list of saved workouts and common genres for user to choose for playlist curation, and save the chosen workout and genre in temporary secondary memory.	Get Request Set Request
Music Playlist ↔ MusicCurate	MusicCurate will match songs based on the selected workout chosen and the user's genre preferences.	Get Request Send Result
Music Playlist ↔ MusicSend	MusicStore will store the playlist and notify the user of playlist creation.	Send Result
Emergency ↔ CreateList	Emergency service will help user create a list of emergency contacts	Get Request Set Request Send Result
Emergency ↔ Report	Receive a signal from the user and notify the emergency contacts.	Get Request Send Result
Emergency ↔ CheckStatus	A report the user's health status will be analysed and potentially sent to their emergency contacts	Send Response
Emergency ↔ LocationStatus	User's location will be pulled from the phone's location/server and then saved and added to the alert message being sent.	Get Request Send Result

Attribute Definitions

Responsibility	Attribute	Concept
R1: Set the users free time interval and workout time	SetFreeTime	Calendar
R2: Calculate and keep track of users free time	CalcFreeTime	Calendar
R3: Send notifications to work out during enough free time	NotifyFreeTime	Calendar
R4: Snooze/Dismiss/Acknowledge free time notifications	FreeTimeResp	Calendar
R5: Enable/Disable free time reminders	ToggleFreeTime	Calendar
R6: Gather data from Google Calendar	SyncCalendar	Calendar
R7: Calculate user's calorie intake/outtake ratio	CalRatio	Calorie Tracker
R9: Recommend exercises/ meals to meet goal	CalDisp	Calorie Tracker
R9: Recommend exercises/ meals to meet goal	FoodRec & ExerciseRec	Calorie Tracker
R10: Compile most frequent exercise/meal list	FreqList	Calorie Tracker
R11: Display list of common exercises for selection and additional options for each exercise post-selection	ExerSelect	Workout Selector
R12: Compiles and saves workouts under user-chosen name	ExerSave	Workout Selector
R13: Prompts user for 3rd-party music subscription login and permissions	MusicSync	Music Playlist
R14: Accesses user-intended workout when user prompts creation of playlist	WorkChoose	Music Playlist

R15: Matches songs to each exercise intensity/BPM and length and compiles into playlist	MusicCurate	Music Playlist
R16: Saves playlist under unique name for later use	MusicSend	Music Playlist
R17: Allows the user to create a list of close contacts for emergency	CreateList	SOS
R18: Allows the user to notify the close contacts list in case of emergency	Report	SOS
R19: Checks the status of the user's health	CheckStatus	SOS

Traceability Matrix

Domain Model	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC 10	UC 11	UC 12	UC 13	UC 14	UC 15	UC 16	UC 17
Calendar	x	x	x	x	x	x											
Calorie Tracker							x		x		x	x					
Workout Notifications					x												
Workout Selector								x									
Music Playlist								x					x	x	x	x	x
SOS Feature										x							

System Operation Contracts

Contract CO1:

Operation:	Exercise Reminder/Indicator
Cross Reference:	User Case UC- 5 : Acknowledge/Snooze Reminders
PreConditions:	User has inputted the time intervals in which they have free time.
PostConditions:	Reminders are stopped until the next free time interval (Acknowledged) or another reminder is sent after 30 minutes if there is enough time available for exercise (Snooze).

Contract CO2:

Operation:	Search Calories Consumed
Cross Reference:	User Case UC - 7: Calorie Search
PreConditions:	User finished a meal and wants to know how many calories to input.
PostConditions:	Calorie tracker records the number of calories taken in.

Contract CO3:

Operation:	Create Personalized Music Playlist
Cross Reference:	User Case UC- 13: App-Generated BPM Playlists
PreConditions:	User requests personal playlist creation and selects a saved workout and genre.
PostConditions:	Music Playlist feature curates playlist, saves it, and notifies the user.

Contract CO4:

Operation:	Danger Notifier
Cross Reference:	User Case UC- 10 : Emergency Responder

PreConditions:	User has premade emergency contact list set up and location is turned on.
PostConditions:	User's device information sent to people on the list.

Project Size Estimation (Use Case Points)

Unadjusted Actor Weight

Actor	Relevant Information	Complexity	Weight
User	User is interacting with the system	Complex	3
Database	Database stores data in the system that is either manually entered or fetched internally.	Average	2
System	System displays the different features the Users can choose from, display results, and send reminders.	Average	2

$$\text{UAW} = (\# \text{ of Simple Actors} \times 1) + (\# \text{ of Average Actors} \times 2) + (\# \text{ of Complex Actors} \times 3) = 2 \times 2 + 1 \times 3 = 7$$

Unadjusted Use Case Weight

Use-Case Name	Relevant Information	Category	Weight
Free Time Notifications (UC#1)	System has to provide users with a notification pop-up. It only uses one actor.	Simple	5
Set Free Time (UC#2)	User will be able to set their minimal amount of free time needed. This requires two actors.	Complex	15
Dismiss Reminders (UC#3)		Average	10
Toggle Notifications (UC#4)	Toggle free time notifications	Average	10

	ON/OFF.		
Acknowledge/Snooze Reminders (UC#5)	Encourages the user to stick to their goal of exercising.	Simple	5
	Waits 30 minutes before checking if the amount of free time is enough to implement a work out.		
Calendar Sync (UC#6)	Allows users to seamlessly add their already created calendars to their device.	Average	10
Calorie Search (UC #7)	Lets users easily add the calories eaten into the system.	Complex	15
Exercise Selector (UC #8)	System will receive that information and know how many calories were burned based on pre-assigned values.	Complex	15
	System will determine the number of calories burned based on the user's past exercise experiences		
Calorie Notifier (UC #9)	Notifies users of the calories lost or gained at the end of each day.	Average	10
Emergency Responder (UC #10)	The system will have a list of people made by the user, and the system will notify those people if the	Average	10

	user presses a button		
	The system will be able to identify user's health emergencies based on their heartbeat and notify the select list of people if need be.		
Professional Recommendations (UC #11)	The system will have a database of healthy recommendations suggested by professionals.	Complex	15
Favorite Meals (UC #12)	App will display the list of favorite meals under the meal search bar.	Simple	5
App generated BPM playlists (UC #13)	App will display the selected playlists and the user will have the option to save them for future workouts	Complex	15
Workout rewards (UC #14)	The app will then reward the user, in terms of subscriptions of one month	Average	10
User preferred playlists (UC #15)	The app will also generate playlists based on the artists the user likes/prefers and will allow the user to save them	Complex	15
Playlist notifications (UC #16)	The app will generate new playlist notifications for the user, and perhaps in conjunction with the workout reminders to increase the	Simple	5

	motivation factor.		
Heart rate and BPM sync (UC #17)	The app will change songs/ music tempo by detecting the user's heart rate, high BPM songs will play	Complex	15

UUCW = (Total # of Simple Use Cases x 5) + (Total # of Average Use Cases x 10) + (Total # of Complex Use Cases x 15) = 4*5 + 6*10 + 7*15 = **185**

UUCP = UUCW + UAW = 185 + 7 = **192**

Technical Complexity Factors (TCF):

Technical Factor	Description	Weight	Perceived Complexity	Calculated Factor
T1	Distributed System (app version and watch version)	2.0	3	6
T2	Users expect a good response time	1.0	3	3
T3	End-user expects efficiency but nothing exceptional	1.0	4	4
T4	Internal processing is complex	1.0	4	4
T5	Code Reusability	1.0	1	1
T6	Ease of install is important	0.5	3	1.5
T7	Ease of use is very important	0.5	4	2
T8	Portability to other platforms	2.0	2	4
T9	Somewhat difficult to modify or add new features	1.0	3	3
T10	Concurrent use (by multiple users)	1.0	3	3

T11	Special security features	1.0	2	2
T12	No direct access for third parties	1.0	0	0
T13	No special user training facilities required	1.0	0	0

Technical Factor Total: 33.5

$$TCF = C1 + C2 * \text{Technical Factor Total} = 0.6 + (0.01) * 33.5 = \mathbf{0.935}$$

Environmental Complexity Factors (ECF):

Environmental Factor	Description	Weight	Perceived Impact	Calculated Factor
E1	Some familiarity with the UML-based development	1.5	1	1.5
E2	Beginner familiarity with application problem	0.5	2	1
E3	Some knowledge of object-oriented approach	1	3	3
E4	Beginner lead analyst capability	0.5	2	1
E5	Highly motivated	1	4	4
E6	Stable requirements expected	2	3	6
E7	No part-time staff involved	-1	0	0
E8	Using Java, some familiar with it and others willing to learn	-1	3	-3

Environmental Factor Total: 13.5

$$ECF = C1 + C2 * \text{Environmental Factor Total} = 1.4 + (-0.03) * 13.5 = \mathbf{0.995}$$

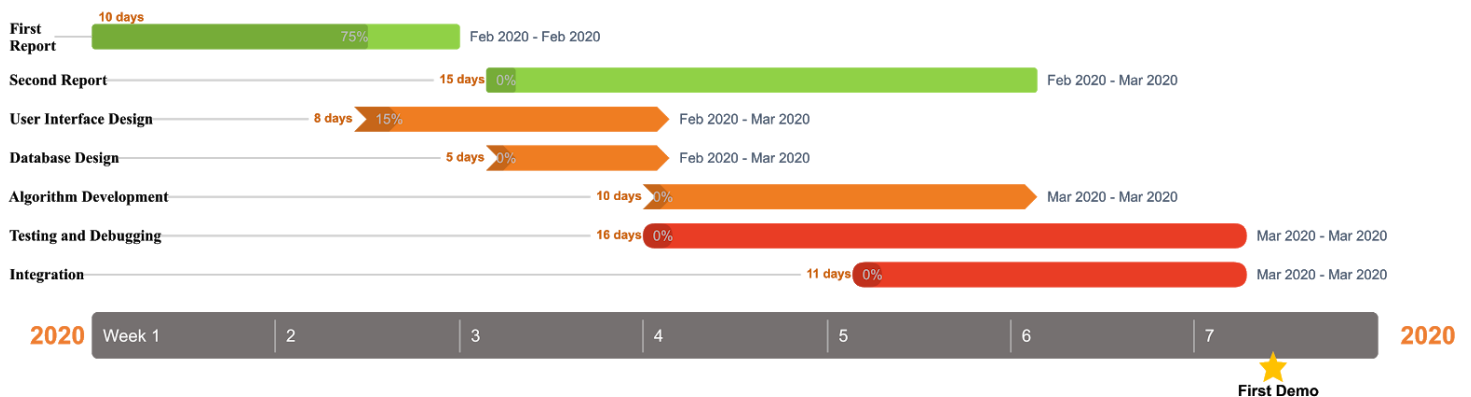
Overall Project Size Estimation:

$$UCP = UUCP \times TCF \times ECF = 192 * 0.935 * 0.995 = \mathbf{178.62}$$

Project Management

Project Road map

Until First Demo



References

1. "FitBit Health Monitoring Analytics"
<https://www.ece.rutgers.edu/~marsic/books/SE/projects/HealthMonitor/2014-g5-report3.pdf>
2. "Software Engineering book", Ivan Marsic
http://www.ece.rutgers.edu/~marsic/books/SE/book-SE_marsic.pdf
3. "Learn about the four different types of exercise"
<https://www.captel.com/2013/10/learn-four-different-types-exercise/>
4. "Creating a watchOS App"
<https://developer.apple.com/tutorials/swiftui/creating-a-watchos-app>
5. "3 Music Apps that will create a BPM-based playlist for your workout"
<https://www.mensjournal.com/health-fitness/3-music-apps-will-create-bpm-based-playlist-your-workout/>
6. "Use Case Points"
https://en.wikipedia.org/wiki/Use_Case_Points