Exercise Break App

Prepared by

Ash Kandari, Ben Deschand, Soham Pradhan, Nilesh Jain

CS 440

at the

University of Illinois Chicago

February 2022

Table of Contents

List of Figures

List of Tables

I Project Description

1 Project Overview

The idea of this app is to use the heartbeat sensor technology found in most smart watches to calculate and find the optimal time to take a break and either get a drink or use its GPS to find a place that could sell a sports drink. This way the user takes a break at the optimal time on their bike ride or run to take a break to steady their heart rate and rehydrate during the trip.

2 The Purpose of the Project

2a The User Business or Background of the Project Effort

Clients for our project don't necessarily belong to any business or specific background. The application is designed for people who would like to make their exercise less strenuous.

2b Goals of the Project

This app was made with the intention of helping endurance trainers to those who just want to go on jog to have a better exercise experience. The hope is by making endurance exercise more fun and less strenuous, they will feel more inclined to do these types of exercises and foster healthier habits.

2c Measurement

After using the application the users would notice significant improvements in their enthusiasm for cardiovascular exercise. They would be able to feel more relaxed and inclined to exercise again. The user would be able to see a more stable chart for their heart rate during the exercise which would signify efficacy of the application.

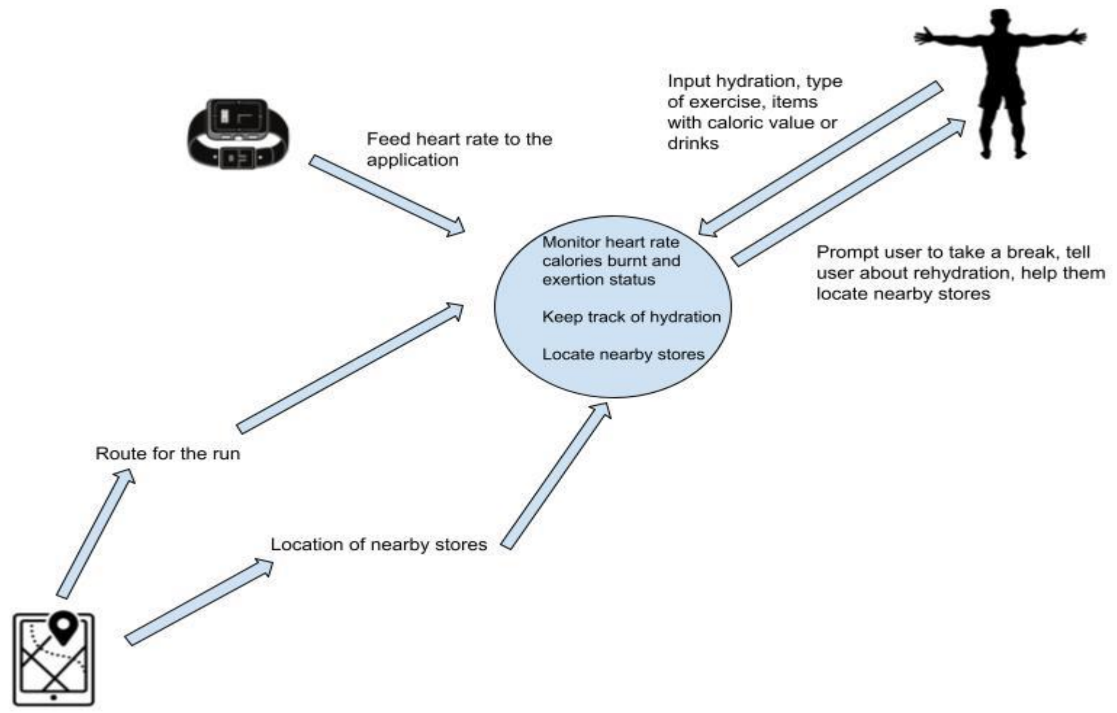
3 The Scope of the Work

The work is making endurance exercises easier for everyone.

3a The Current Situation

The client is currently using other work out apps or none at all when going on endurance exercises. The current workout apps usually track calorie consumption and the routes they took. Some monitor heart rates during the exercise but provide no information on when to take a break and how to space out their run.

3b The Context of the Work


Input hydration, type of exercise, items with caloric value or drinks
Feed heart rate to the application
Monitor heart rate calories burnt and exertion status
Keep track of hydration
Locate nearby stores
Prompt user to take a break, tell user about rehydration, help them locate nearby stores
Route for the run
Location of nearby stores

3c Work Partitioning

Event Name	Input and Output	Summary
Exercise Information	Input: User inputs the type of exercise Ex. Bike, jog	The application will take the type of exercise and determine how energy spending the exercise would be. Average value of exertion dependent on the exercise
Hydration Status	Input: User inputs the amount of water they drank before the exercise	The application will use the input to keep track of hydration of the user
Calorific Value (items) for the exercise	Input: the user inputs what snack or drink they would be bringing for the exercise	This helps the application keep track of the caloric value that user has with them
Route for exercise	Input: User inputs their route for the exercise	Application keeps track of what route the user wants to take to locate stores on the path

Heart Rate (Smart watch)	Input: heart rate to the application	Application takes the heart rate from the watch and monitors it. Checks where the heart rate is within the spectrum of exertion
Break	Output: The application sends out and alert with a soft vibration to the user	The application uses the heart rate being sent by the smart watch and when it crosses a relative threshold for the user it prompts the user to take a break.
Hydration Break	Output: Soft vibrate to the phone for hydration	The application reads when the hydration status goes below a specific level and alerts the user to rehydrate
Locate Store(Sub event of Hydration Break)	Input: prompt the user to locate stores Output: provide the location of nearby stores	The application prompts the user if they want to look for a store and if answered yes it looks for nearby stores and fetches it and provides it to them

3d Competing Products

There are many competing products out their with these 5 being considered the top ones:

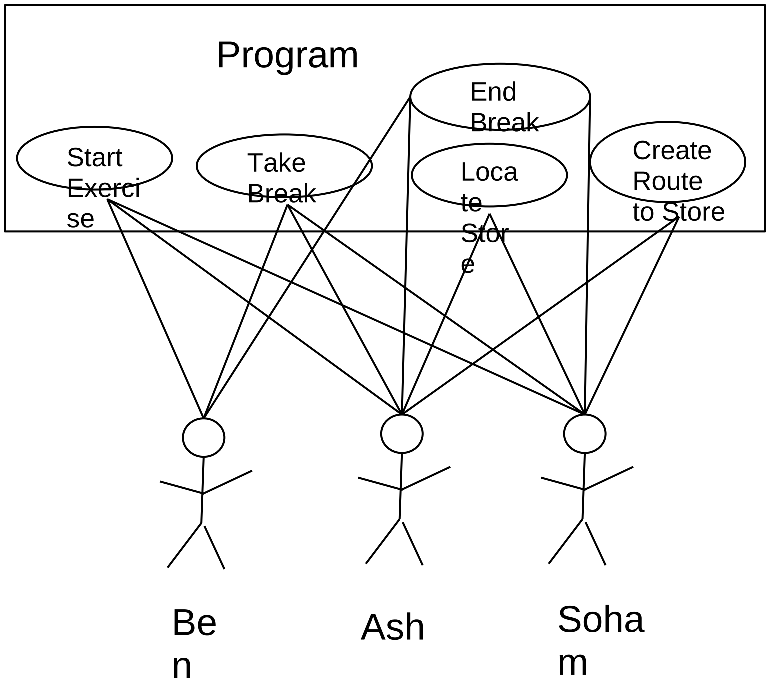
- Map My Run
    - Has millions of routes to try and records the distance traveled, calories burned, and more
- Adidas Running App by Runtastic
    - Easy tracking and stores your run data
- Pumatrac
    - Uses outside information and previous runs to give insight on the perfect time to run
- Nike Run Club
    - Allows you to share your personal goals with friends
- Couch to 5k
    - Made to help beginners start running

While these products do complete a number of different tasks that are important to those going on endurance exercises, they do not have the two main features that our product has. None of the products shown above will alert you to take breaks if your heart rate reaches a certain number and, if needed, a route to the nearest store.

4 The Scope of the Product

The product will help those who are participating in endurance exercise take breaks when their heart rate is too high and makes sure they are properly hydrated while on this exercise.

4a Scenario Diagram(s)



4b Product Scenario List

Name	External Actors	Relevant Information
Simple Run and Break	Ben	Has never used the app.
Bike with Route to Store for Break	Ash	Has used the app before and has already created a profile.
Run with a Break and then Second Break at Store	Soham	Has used the app before and has already created a profile.
Run with Breaks	Nilesh	Has never used the app.

4c Individual Product Scenarios

Simple Run and Break: Ben decides to go on a run. He downloads the app from the app store of his choosing and also the requisite app for his smart watch. He creates an account and inputs information such as his height, weight, age, and his level of expertise with endurance exercises. When he is about to go on said run, he will open the app on his phone. He will select that he is going on a run. Since he is bringing a sports drink with him, he will input that into the watch as well. Then he will start his run. During his run, when his heart rate is above the healthy amount for his age, his smartwatch will gently buzz alerting him to take a break. Ben will then find a safe place to stop and press a button on his smartwatch screen to start the break. He will then consume his energy drink and relax while the smartwatch monitors his heart rate. Once it reaches a certain level, it will buzz him to let him know that he can start again. It will ask if he consumed his sports drink and he will reply with yes. Ben will then run home and, once there, alert the program that he has finished.

Bike with Route to Store for Break: Ash wants to go on a bike trip. He has already downloaded the Exercise Break App and created an account. He will say that he is about to go on a bike ride. Since he is not bringing any drink with him, he will input that he does not have a drink. He will then start the exercise on his watch and begin biking. When his heart rate starts to get close to overexertion, the program will start looking for the closest store. It will ask if he wants to stop at the store and Ash will respond that he does. It will then alert Ash, on his smartwatch, and show him the route to the nearest store. Ash will then bike to the store. When he gets there he will start the break on his watch and proceed to buy a drink. Once his heart rate has lowered to an acceptable level, the smartwatch will alert him that he can begin again. It will ask if he got a drink, what he got, and if he finished it. Ash will enter that he had a water and finished it. He will then bike home. When he gets there he will alert the program that he has finished.

Run with a Break and then Second Break at Store: Soham decides he wants to go on a run. He has already downloaded the app and created his account. He will select that he wants to run. Since he is bringing water with him, he will input that he is bringing a water. He will then start the exercise on his smartwatch and start running. Once he reaches a certain heart rate, the smartwatch will buzz, alerting him to take a break. Soham will find a safe spot to rest and tell the program that he started his break. He will rest and consume his water until the watch tells him that he can start running again. He will alert the program that he has finished his drink. He will then continue running. Once his heart rate starts to climb to the overexertion level, it will now look for the nearest store since he ran out of drink. It will ask if Soham wants to stop at a store and he will say he does. It will then show him a route to the nearest store. Once he gets there, he will start the break. He will grab a sports drink and relax to lower his heart rate. It will alert Soham that he can start running once his heart rate reaches an acceptable level. It will ask if he got a drink, what he got, and if he finished it. Soham will reply that he got a sports drink and drank it. He will then run home. Once he gets there, he will tell the program that he has finished.

5 Stakeholders

5a The Client

We, the organization that is creating the application, will fund our development of the Exercise Break App, acting as a client.

5b The Customer

The application is targeted at the general public who wants to participate in endurance exercise, but can also be used for professional purposes like for athletes and their training regime.

5c Hands-On Users of the Product

The application is not targeted to any organization which makes it hands-on to the customers themselves. Since the application is made for exercise it is not restricted to any specific category. The user should be able to use smartphones and connect their monitoring device.

5d Maintenance Users and Service Technicians

The users will first have to download the application. Our team will update the software and will be observing the functionality of the application. If it’s glitched or does not work properly, then the development team would look over the issue.

5e Other Stakeholders

The application at first seems to only have one stakeholder but the application can be used by organizations that plan marathons and other exercise events. The organization would benefit from the application as it would help them to mandate the events better in relevance to the runners health. The involvement of these stakeholders would require changes to application so that it would fit the event.

5f User Participation

There is participation required by the user to maintain the efficacy of the application. Input data such as hydration status, deciding a route and calorific value are necessary for the application. Since the application is designed to work side by side with the exercise it would take about 2 to 3 minutes to input this data. The application involves notifying the user about breaks and prompting them for other things which makes it a back and forth relationship with the user.

5g Priorities Assigned to Users

Key Users: The key users are the people using the application to benefit their exercise experience and make it more enjoyable. The success of the application primarily depends on them.

Secondary Users: Secondary users involved would be the organizations responsible for organizing marathons, even though these users benefit from the working of the app they are not directly affected by its use. There are also users who would want to use the application for its other features such as hydration status so satisfying these users is not a big priority.

6 Mandated Constraints

6a Solution Constraints

This app requires the use of a smart watch with a heart beat sensor.It also assumes that the watch would be able to connect to the smartphone via Bluetooth/Internet and give the user appropriate response.

6b Implementation Environment of the Current System

The proposed product will operate on both iOS(WatchOS) and Android(WearOS).

6c Partner or Collaborative Applications

This application must be able to fetch data from Google maps to indicate nearby routes and shops. It is also responsible for fetching the heart beat data from the heart rate sensor every second.

6d Off-the-Shelf Software

The application contains COTS software such as Google/Apple Maps, for determining the routes and shops where drinks would be available for users.

6e Anticipated Workplace Environment

The product will be used mostly in environments suitable for walking/jogging/running.

6f Schedule Constraints

The product will be more accurate and useful when the app would be able to detect heartbeat sensors more frequently. It is least beneficial if the user tends to remove the watch during the runs.

6g Budget Constraints

This application can be managed by 5 engineers. Since it is a software, there are no additional hardware costs incurred. We estimate a budget of around $1M including all the advertisement costs.

7 Naming Conventions and Definitions

7a Definitions of Key Terms

Heart Beat Sensors: These are essentially pulse monitors worn on the wrist that requires for a person to be in constant contact with the sensor which takes a person's pulse and uses that to monitor heart rates.

Hydration: Hydration is the process of replacing water in the body. This can be done through drinking water, eating ice chips, eating foods that have high water content, and drinking other fluids.

Exertion: Exertion is a state of mental and physical discomfort(for our application it's physical)

Calorific Value: Hydration is the process of replacing water in the body. This can be done through drinking water, eating ice chips, eating foods that have high water content, drinking other fluids

7b UML and Other Notation Used in This Document

The diagram used in 3b uses standard symbols and notations for explaining the input, process and output procedures. The arrows are used to show the direction of information flow.

7c Data Dictionary for Any Included Models

The maximum heart rate is calculated with the following formula:

200- Age of the person = heart rate in BPM

The heart rate dependent on the exercise are calculated with the following formula:

For moderate intensity exercise(the heart rate ranges between 64% to 76% of max heart rate)

for 64%: max heart rate * 0.67

76%: max heart rate * 0.76

For vigorous intensity exercise(range is in between 77% to 93%)

for 77%: max heart rate * 0.77

93%: max heart rate * 0.93

8 Relevant Facts and Assumptions

8a Facts

The heart rate spectrum for the application works on statistical calculations that take in the intensity of the exercise and estimated heart rate dependent on the user's physique and age.

8b Assumptions

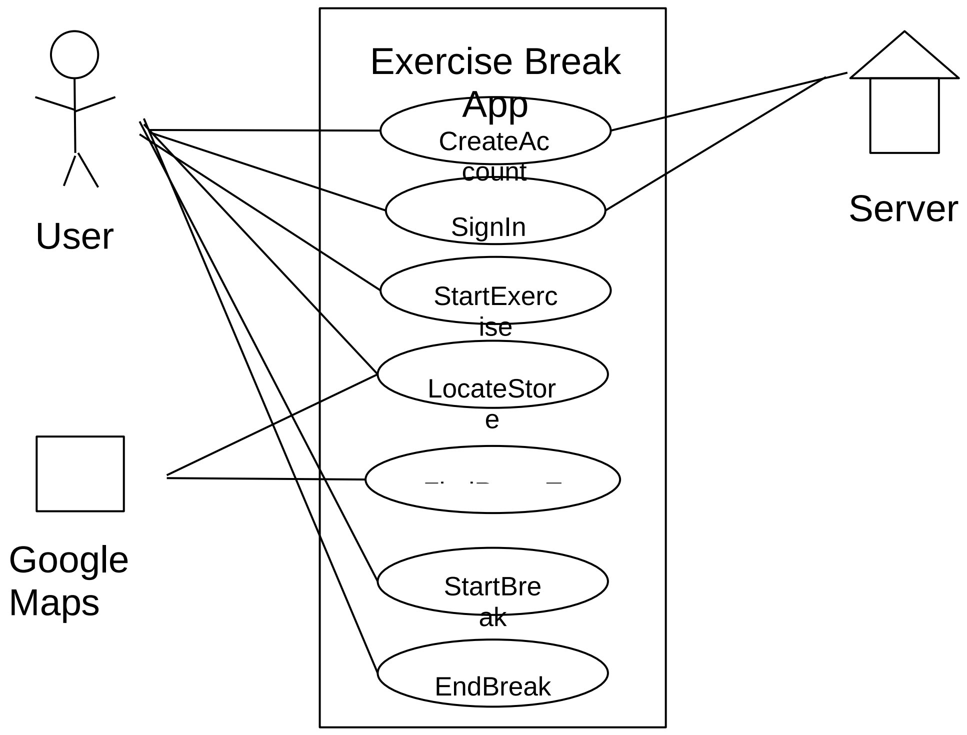
The application is not capable of understanding fluctuations in the heart rate that are caused by other factors such as psychological stress or an underlying health condition. So we assume that the user is medically and emotionally fit.

We are also assuming that the user has access to a smart watch or any other heart beat monitoring device that can be connected to the smartphone.

II Requirements

9 Product Use Cases

9a Use Case Diagrams



9b Product Use Case List

ID	Name
UC-1	User-CreateAccount
UC-2	User-SignIn
UC-3	User-StartExercise
UC-4	User-LocateStore
UC-5	User-StartBreak
UC-6	User-EndBreak
UC-7	Maps-FindRouteToStore

9c Individual Product Use Cases

Use case ID: UC-1 Name: User-CreateAccount

pre-conditions: Program must be installed

post-conditions: Account created and sent to server and User sent to home page

Initiated by: User

Triggering Event: User pressing “Create Account” button

Additional Actors: Server

Sequence of Events:

1. User presses the “Create Account” button

2. System moves to sign up page

3. User inputs email and creates password

4. System moves to additional information page

5. User inputs information to produce BMI (height, weight, age, etc.) and presses the “Submit” button

6. System sends the User to the user home page and the account information is sent to a server

7. Server sent user information to store

Alternatives: NA

Exceptions: User inputs invalid email

Use case ID: UC-2 Name: User-SignIn

pre-conditions: User already created account

post-conditions: User sent to home page

Initiated by: User

Triggering Event: User pressing “Sign In” button

Additional Actors: Server

Sequence of Events:

1. User presses the “Sign In” button
2. System moves to sign in page
3. User inputs email and password and presses the “Submit” button
4. System sends information to server
5. Server checks for email and password and returns the User information
6. System gathers User information and moves to user home page

Alternatives: User inputs incorrect email or password

Exceptions: NA

Use case ID: UC-3 Name: User-StartExercise

pre-conditions: User already created account

post-conditions: System actively monitoring heart rate

Initiated by: User

Triggering Event: User pressing “Start Exercise” button

Additional Actors: NA

Sequence of Events:

1. User presses the “Start Exercise” button
    2. System moves to the exercise information page
3. User inputs the type of exercise and what they are bringing to hydrate
    4. System stores that information to be used during exercise
5. User presses the “Start” button
    6. System starts monitoring the User’s heartbeat

Alternatives: NA

Exceptions: NA

Use case ID: UC-4	Name: User-LocateStore
pre-conditions: User must have started the exercise, have a heart rate that is close to the requirements for needing a break, and didn't bring anything to hydrate post-conditions: Google Maps looks for a store nearby Initiated by: User Triggering Event: User pressing "Locate Store" button Additional Actors: Google Maps	
Sequence of Events: 1. User presses the "Locate Store" button 2. System contacts Google Maps 3. Google Maps uses User's current location to find the nearest store that would sell drinks to hydrate.	
Alternatives: NA Exceptions: User does not have internet connection	

Use case ID: UC-5	Name: User-StartBreak
pre-conditions: User must have started exercise and their heart rate must reach a certain point. post-conditions: User starts break Initiated by: User Triggering Event: User pressing "Start Break" button Additional Actors: NA	

Sequence of Events:

1. User presses the “Start Break” button

2. System moves to the break page and monitors the User’s heartbeat.

Alternatives: NA

Exceptions: NA

Use case ID: UC-6 Name: User-EndBreak

pre-conditions: User must have started a break

post-conditions: User ends break

Initiated by: User

Triggering Event: User pressing “End Break” button

Additional Actors: NA

Sequence of Events:

1. User presses the “End Break” button

2. System moves end break page

3. User inputs what they drank and if they finished it and presses the “Submit” button

4. System moves back to the exercise page and continues to monitor the User’s heartbeat

Alternatives: NA

Exceptions: NA

Use case ID: UC-7 Name: Map-FindRouteToStore

pre-conditions: User presses the “Locate Store” button

post-conditions: Google Maps sends route to system

Initiated by: System

Triggering Event: System sends User’s location

Additional Actors: Google Maps, User

Sequence of Events:

1. Google Maps looks at the location of the store and the User’s location

2. Google Maps creates route between two and sends route to System

4. System receives route and displays it for User

Alternatives: NA

Exceptions: User has no internet connection

10 Functional Requirements

F1 - Creating an Account

Description: The User must be allowed to create an account that is identified by an email and password. They must then be able to put in their age, height, and weight so the program can find the overexertion heart rate. This information will be stored on a server.

Rationale: The information will be used to find the correct heart rate to stop for a break for each person. Storing the information on a server will allow the User to access their account on a different device, should they get a new one.

Fit Criterion: The User’s data is correctly stored in the server.

Acceptance Tests: T1, T7

F2 - Logging into an Account

Description: The User must be able to get their information from inputting an email and password. This should get the information from the server.

Rationale: This allows the User to use multiple devices with the same account.

Fit Criterion: The User’s data is correctly found in the server.

Acceptance Tests: T1, T7

F3 - Starting an Exercise

Description: The User picks what drink they are bringing and what type of exercise it is and then starts the exercise on the program, alerting the program to start watching their heart rate.

Rationale: The program needs to know when the User is exercising and what they are bringing to drink so the program can know when to start observing the User's heartbeat and alert them to take a break, or find a store if they don’t have anything to drink.

Fit Criterion: The program alerts the User that they started their exercise and it will eventually notify the User to take a break.

Acceptance Tests: T2

F4 - Starting a Break

Description: During the time an exercise is initiated, when the User’s heartbeat reaches the calculated rate, the program alerts the User that they have a high heart rate and should take a break. The User should then be able to start the break on the program, so the program can watch the User’s heart rate to alert them to end the break when it lowers to a calculated lower limit.

Rationale: The program needs to alert the User to take a break, while the User needs to alert the program that they have started a break.

Fit Criterion: The break is started at the correct heart rate.

Acceptance Tests: T2

F5 - Ending a Break

Description: When the program finds the User's heart rate has gone down to an acceptable range, it alerts the User to start their exercise again. It will also ask if they finished their drink.

Rationale: The User needs to know if they are ready to start their exercise again, according to their heart rate. The program needs to know if the User still has something to drink in case it needs to find a store for the next break.

Fit Criterion: The program notifies the User in the correct heart rate range. The User inputs they used their drink and the program notifies the User to allow it to locate a store near the next break.

Acceptance Tests: T2

F6 - Locating a Store

Description: The program should be able to access Google Maps to find the closest store to the User according to their current location.

Rationale: This is needed when the User needs to find a store near them to buy a drink during their break.

Fit Criterion: The closest store was correctly found near the User.

Acceptance Tests: T2

F7 - Creating a Route

Description: The program accesses Google Maps to find the best route between the User's location and the closest store.

Rationale: The route is needed to alert the User to the fastest way to the store so they can get there quickly and start their break.

Fit Criterion: The route to the store is the shortest route between the User and it.

Acceptance Tests: T3

F8 - Monitoring a Heartbeat

Description: The program must be able to observe the User's heartbeat through the device they are wearing, like a smartwatch.

Rationale: This data is needed to compute the User's heart rate so the User can be alerted to take a break.

Fit Criterion: The program's calculated heartbeat is within an acceptable range.

Acceptance Tests: T3

11 Data Requirements

D1 - User information

Description: The program will store the User's height, weight, and age into a server.

Rationale: This will allow the program to calculate the correct BMI and find a personalized heart rate that would be considered in the range of overexertion.

Fit Criterion: The program finds the correct BMI for the User based on the information received.

Acceptance Tests: T1, T4

D2 - Email

Description: The email the User puts in must be properly formatted, with the At sign and a period found.

Rationale: These are the two parts needed for a functioning email so both must be found to create an account.

Fit Criterion: The program accepts and declines the appropriate emails.

Acceptance Tests: T1, T4

D3 - Password

Description: The password must be eight characters long and have one number and one uppercase letter.

Rationale: This will make the password more secure for the account.

Fit Criterion: The program accepts and declines the appropriate passwords.

Acceptance Tests: T1, T4

12 Performance Requirements

12a Speed and Latency Requirements

P1 - Latency for Account Creation and Break Calculation

Description: The accounts are linked to an online server so the server to UI response time should be under 3 seconds

Calculation when it comes to Optimal break in the exercise should almost be instantaneous because the calculation is done locally

The application will poll the sensor at every 0.5 seconds to keep an accurate measurement of heartbeat

Rationale: The UI to server response time has to be under 3 seconds so the user so that the application can be used instantly

Calculation is not mathematically heavy and the notify the user for the break takes priority in the application and hence need to be done instantaneously

The break time and its calculation is linked to the monitoring of the heartbeat and to provide continuous data the application needs to be fed heartbeat continuously.

Fit Criterion: N/A

Acceptance Tests: P19

12b Precision or Accuracy Requirements

P2 - Accuracy of calculations

Description: The precision with which the application decides to prompt the user for a break should be down to every second

Accuracy for heartbeat should be within ± 3 bpm

The distance for nearby stores should be within ± 10 meters

Rationale: Since the application is to reduce exertion the break needs to be precise down to every second.

Heartbeat being fed to the application can vary within ± 3 bpm, as anything more or less than that would make it harder for the application to put in range for the spectrum(exertion level) and make it inaccurate.

Fit Criterion: The device is being fed heartbeat constantly and is always accounting for the margin of error values while calculating the break time.

Acceptance Tests: P4

12c Capacity Requirements

P3 - Maximum Users

Description: Maximum number of users to be dealt with at a time is 1.

Rationale: Since the application is based on one on-hand user the capacity requirement does not really apply.

Fit Criterion: Input values to the application for heartbeat should be inputs from only 1 user

Acceptance Tests: P4

13 Dependability Requirements

13a Reliability Requirements

DP1 - Reliability

Description: The product is not allowed to fail in any situation.

Rationale: The application is not allowed to fail in the duration of the exercise. Most of the calculations are done locally so there would be no reason to fail. Since most of the data is input and output from either sensors or other applications. The system is likely to never crash

Fit Criterion: Making sure that there are no fails during the run, and if there are ensuring that they are not from computation within the application or from any of its features.

Acceptance Tests: T6

13b Availability Requirements

DP2 - Available runtime

Description: The product will be available to use 24/7 for 365 days

Rationale: The availability of the application is not dependent on the maintenance of the servers as the application runs primarily on the local devices. The data for the application to run is saved both on the device and the server so in case of maintenance the application uses the local data to run.

Fit Criterion: Program is running 24/7.

Acceptance Tests: T6

13c Robustness or Fault-Tolerance Requirements

DP3 - Robustness

Description: The application will run its primary feature of determining an optimal break time if the application loses access to the internet.

Rationale: The application does not necessarily rely on internet connectivity, so the application should still be functional for its primary feature.

Fit Criterion: Checking if the heartbeat provided by the device is being used to calculate an optimal break time (without any interaction with online services)

Acceptance Tests: T5, T6

13d Safety-Critical Requirements

DP4 - Safety Issues

Description: The application does not affect harm to health, safety, or property in any means

Rationale: The application is made for providing tools to make exercise less strenuous.

Fit Criterion: N/A

Acceptance Tests: N/A

14 Maintainability and Supportability Requirements

14a Maintenance Requirements

M1 - Maintenance

Description: The application primarily relies on the local device implying that even if there is an update to the application old versions would still work on the local device.

Updating the application on the local device is in the hand of the user, for a new version the user will be prompted about the update and they might choose if they want to update at any given time.

Maintenance for the servers that hold the user information should be done by the developers and should be done as early as possible.

Rationale: Since the application is made for personal use and can be used offline, it is essential that hand on users are not affected by any maintenance in real time.

Maintenance of the servers for the user information does not affect prior users but for new users their information is saved both locally and on the server. If trying to access their profile on a different device it is essential that the servers must be running to access that information.

Fit Criterion: Checking if the data is being constantly saved and updated locally so that computations are done even if the application is not connected to the server. For users trying to access their old account, the application is temporarily asking for basic information so the run can be carried out.

Acceptance Tests: T7

14b Supportability Requirements

M2 - Support

Description: The application is self supporting with all necessary information being provided within it.

How to use and other features will be shown in the application with a help tab assigned to the application which will contain all the necessary information.

The application does not have a help desk or any live assistance, this is because of the nature of the application.

Rationale: Since the application is able to run in offline mode, it is made in such a way that least assistance is required. The help tab keeps all the information needed to help in case of any inconvenience.

Fit Criterion: The support for every inconvenience can be found within the application.

Acceptance Tests: T7

14c Adaptability Requirements

M3 - Adaptability

Description: The application is primarily designed for smartphones so the platforms include Android and iOS.

In case the application is to be used by marathon and other event managements the application might be made compatible with Windows or MacOS

Rationale: The primary purpose of the application is to assist the user to take a break and that is why it will be deployed to smartphone's OS which include Android and iOS.

Managing a lot of people for a marathon requires a lot more features and for the processing and easier management it might be deployed to MacOS or Windows in the future.

Fit Criterion: Application will detect the OS and will perform accordingly.

Acceptance Tests: T8

14d Scalability or Extensibility Requirements

\ M4 - Extensions

Description: Product is initially designed for keeping track of one person's heartbeat and their level of exertion, but in case of deploying it for an organization it will be expanded to take care of upto 200 users. There will be more features added to the application which would user other sensors such as sweat pads and oximeter to keep tracks of other vitals

Rationale: The application is focused and will be popular for its one feature which helps users to take a break in case of exertion, but to broaden the horizons there will be new features developed which will take other cases such as low oxygen in account.

Fit Criterion: Ensuring that the application is keeping track of multiple users and outputting their exertion rate and estimated break time.

Acceptance Tests: T8

14e Longevity Requirements

M5 - Longevity

Description: The product is expected to run for minimum of 10-15 years in accordance with maximum maintenance budget

Rationale: The application requires minimum maintenance because of the nature of the app which uses local devices for its primary use and the industry for the application is not a dying industry since fitness and running are going to be a practice for many years to come.

Fit Criterion: N/A

Acceptance Tests: T8

15 Security Requirements

Special protection is to be done to preserve users privacy such as the heart rate which is feeded into the system.

15a Access Requirements

S1 - Access

Description: We propose 2 types of security access, one for the user and second for the developer.

Rationale: The user will be able to get access to all the map routes and nearby stores where they would be able to get refreshments. A developer would get access to change the UI as well as the back end of the application.

Fit Criterion: The application gives appropriate access to the user as well as the developer.

Acceptance Tests: T9

15b Integrity Requirements

S2 - Integrity

Description: Privacy of user is of utmost importance and it will not be seen even to the developers of the application.

Rationale: The user will be able to securely enter their data which would be encrypted and saved in the database. Examples of user data include the user login and password used to login to the application. Developers will be able to make modifications to the schema and change the UI of the application but will not be able to see any private user data.

Fit Criterion: Users and developers are only able to access data which they have access to.

Acceptance Tests: T9

15c Privacy Requirements

S3 - Privacy

Description: Application uses private user data such as users heart beat in order to correctly identify the most optimal time for the user to take a break.

Rationale: Private data such as heartbeat is known only to users. Developers updating the application will have no access to the data. Users data will not be available for public use.

Fit Criterion: Private data of users is known only to the user.

Acceptance Tests: T9

15d Audit Requirements

S4 - Audit

Description: The UI Design will consist of analysis on user data which has been feeded to the application. For example, users will be able to see their average heartbeat during the runs, miles covered during run and calories burnt.

Rationale: Application makes sure that the data which user feeds into the application is audited correctly and displayed to the user.

Fit Criterion: Documentation of user data is done as they use the application.

Acceptance Tests: T10

15e Immunity Requirements

S5 - Immunity

Description: Security of application is paramount. The application is built on major operating systems which provide the feature to protect applications from malware.

Rationale: The application is built on Android as well as ios.Both of these operating systems have virtual sandboxes which are able to avoid most of the viruses, worms and trojan horses. Ios has a default security built in while android needs the feature enabled in order to use it.

Fit Criterion: The application is safe against malware

Acceptance Tests: T9

16 Usability and Humanity Requirements

16a Ease of Use Requirements

UH1 - Ease of Use

Description: The application is meant for people who want to stay healthy irrespective of their age group.

Rationale: The UI of the application is very interactive and is made by professional UI Designers who focus on making a highly interactive UI with high intuitiveness for older generation people.

Fit Criterion: The application is accessible to people of all age groups.

Acceptance Tests: T11

16b Personalization and Internationalization Requirements

UH2 - Personalization

Description: The application will be made available in English. All the payments are to be done in United States Dollar. In the future, the team may accept crypto for in-app services.

Rationale: This application is primarily being designed for the United States where English is the most widely spoken language. English remains the most widely spoken language in the World. In future, the developers can add other languages in the application to address potential users who cannot speak/read English.

Fit Criterion: The application can be used by people worldwide.

Acceptance Tests: T11

16c Learning Requirements

UH3 - Learn

Description: The application is designed by professional UI Designers. It is intended for users of all age groups.

Rationale: In the recent decades, there have been great advances in the ability to make powerful UI's which are incredibly easy to use, eg. getting the user's work done in fewer clicks, adding effects to every click to make everything look aesthetically pleasing. We intend on using these features to make our UI accessible to people of all age groups.

Fit Criterion: The application is easy to learn.

Acceptance Tests: T12

16d Understandability and Politeness Requirements

UH4 - Understandability

Description: The application will give a guided tour of its application to all the new users. As soon as the new user logs in, the application will first introduce them to all the graphical icons which are being used in the application. As soon as the user clicks on the icon, the application will tell the user about the use case of those icons and when they need to access the icon.

Rationale: Introducing a user friendly guide at start is important for users to fully understand how the application works. If users are able to understand the application fully, they will interact with it more and possibly love using it.

Fit Criterion: Users will be able to go through instructions directly anytime.

Acceptance Tests: T12

16e Accessibility Requirements

UH5 - Accessibility

Description: The application can be used by people with disabilities, not all disabilities are covered by the app, for example deaf people can still use the app without any problem.
Rationale: There are people with a wide range of disabilities and covering all of them is not feasible for the business of this app. The application tries to cover and accommodate the majority of people.

Fit Criterion: Any user can access the application.

Acceptance Tests: T12

16f User Documentation Requirements

UH6 - Instructions

Description: The application will give a guided tour to the user whenever a new user logs in for the first time. A list of documentation will be displayed on the app store when the user can install the application.

Rationale: Users will be introduced to the application fully and all the documentation about the data privacy will be readily available to the user.

Fit Criterion: User will be able to easily access the application because of documentation done through UI

Acceptance Tests: T13

16g Training Requirements

UH7 - Training

Description: Users should be able to operate a smartphone device as well as smartwatch. Users should refer to documentation about the device they are using from the manufacturer.

Rationale: The application contains a program which trains the user about the app, it is assumed that the user is able to operate a smartphone and smartwatch.

Fit Criterion: User will have one-to-one interaction with the application and can go through basic needs that the user has to have to operate the program.

Acceptance Tests: T13

17 Look and Feel Requirements

17a Appearance Requirements

A1 - Appearance

Description: The application will be organized in such a way that the user will be able to easily access what he/she wants. The user can also choose font styles, background colors, themes, etc. according to their preference. This is to make it user friendly. We will have a logo for the App.

Rationale: The application will have choices containing different GUI settings for the user.

Fit Criterion: Users can interact with the program (colors, font, etc.)

Acceptance Tests: T14

17b Style Requirements

A2 - Style

Description: For now, the box will contain a sensor, a charging wire and straps with instructions. This product will be available in different colors. Its up to the user which color he/she prefers. Mostly the youth will get attracted to this product as they are active. To encourage children and senior citizens to buy the product, we will be adding games and peaceful meditation music and videos in the program.

Rationale: The application contains various themes, features, etc. and comes in different colors.

Fit Criterion: User can choose the product according to his/her style.

Acceptance Tests: T14

18 Operational and Environmental Requirements

18a Expected Physical Environment

OE1 - Physical Environment

Description: The user has options to run at a place which is included in the maps. It is recommended to have an Internet connection between smartwatch and smartphone. Sometimes, the watch switches to bluetooth when the smartphone is nearby, but an Internet connection would guarantee the success of transmitting data.

Rationale: Since bluetooth only connects for 10-20 meters, it is advised to have a stable internet connection while on a run. If the user runs with their smartphone, a bluetooth connection should suffice as long as the place they are jogging is included in the maps.

Fit Criterion: The data from the user's activity and location will be saved and then analized by the system. After this, it will provide the user with the best possible routes to run, break spots, etc.

Acceptance Tests: T15

18b Requirements for Interfacing with Adjacent Systems

OE2 - Interface

Description: This application works on the principle of communication between the server database and user. The inputs from the user gets analyzed by the sensor that the user is wearing. This data will be then sent to the server and the program will make decisions accordingly.

Rationale: It is advised that the user buys the whole product (every part) so as to get maximum efficient and successful results. Input and Output messages will be transferred wirelessly between sensor and server (frontend and backend will be connected).

Fit Criterion: User will be able to provide input directly to the sensor and through there it will be transmitted to the database. This will allow the App to make efficient decisions for the user and there will be a strong and direct relationship between these two.

Acceptance Tests: T15

18c Productization Requirements

OE3 - Productization

Description: The application is free to download on Appstore/Google Play. It is recommended to have the latest version of OS on smartphones and smartwatches for smooth functioning of the application.

Rationale: Developers are making the application accessible to people with lower versions of android/ios but for best experience, it is recommended to use the latest version of OS.

Fit Criterion: User has access to Appstore/Google Play to download the application.

Acceptance Tests: T15

18d Release Requirements

OE4 - Release

Description: It is intended that the product should be made available for beta version at first. The developers will then work on the final complete product to be released in the market.

Rationale: Beta version of this application will be released first and will have the most basic commands and controls for the user to operate the App. After this, the full version of the App will be released and the developers will observe the feedback about the program from users and try to nourish it better.

Fit Criterion: Users will have access to updated versions.

Acceptance Tests: T15

19 Cultural and Political Requirements

19a Cultural Requirements

CP1 - Distance Measurements

Description: The program must be able to find the distance and route to a store in the measure of distance common to that area.

Rationale: Different countries use different systems of measurement, which means the program must be able to account for that by showing the correct measurement.

Fit Criterion: The program shows the route in miles if American but meters if British.

Acceptance Tests: T16

19b Political Requirements

CP2 - Color Scheme

Description: The color scheme of the program shall not copy the color scheme of other popular fitness apps.

Rationale: The development team doesn't want the program to look as though it is from a different company, but should feel like it's own unique product.

Fit Criterion: The color scheme is unique against other popular fitness apps found on the store.

Acceptance Tests: T16

20 Legal Requirements

20a Compliance Requirements

SC1 - Compliance

Description: The application will have Copyrights and disclaimer for the user before accepting all the Rules and Regulations of the App. If the developers find out about any other application which uses the same features as the Exercise Break App, they can put legal charges on that person/company.

Rationale: We will take legal approval from the respective authorities to initiate this project and use it for a long period of time.

Fit Criterion: N/A

Acceptance Tests: T17

20b Standards Requirements

SC2- Standards

Description: The development team will develop the program according to the Agile development steps.

Rationale: This will allow the development team to work on and complete workable steps that will allow the code to easily grow.

Fit Criterion: N/A

Acceptance Tests: T18

21 Requirements Acceptance Tests

21a Requirements – Test Correspondence Summary

Table 1 - Requirements - Acceptance Tests Correspondence

	F1	F2	F3	F4	F5	F6	F7	F8	D1	D2	D3	P1	P2	P3	DP1	DP2	DP3	DP4	M1	M2	M3	M4	M5	S1	S2	S3	S4	S5	UH 1	UH 2	UH 3	UH 4	UH 5	UH 6	UH 7	A1	A2	OE1	OE2	OE3	OE4	CP1	CP2	SC1	SC2
T1	x	x							x	x	x																																		
T2			x	x	x	x																																							
T3							x	x																																					
T4									x	x	x		x	x																															
T5																	x																												
T6															x	x	x																												
T7	x	x																	x	x																									
T8																					x	x	x																						
T9																								x	x	x		x																	
T10																											x																		
T11																													x	x															
T12																															x	x	x												
T13																																		x	x										
T14																																				x	x								
T15																																						x	x	x	x				
T16																																										x	x		
T17																																												x	
T18																																													x
T19												x																																	

21b Acceptance Test Descriptions

#T1 - User data storage

Description: The test checks if user data is correctly stored and found in the server

#T2 - User activity alert

Description: The test notifies users of the start and end of an activity/break/store.

#T3 - User tracking check

Description: The test notifies if the user has started on a particular route and is successfully using the application.

#T4 - User signup check

Description: The test checks if initial user information align with username and password is correctly stored

#T5 - User heart monitor check

Description: The test checks the accuracy and latency of heart monitor

#T6 - Application dependability check

Description: The test checks if the application is reliable, if its servers are available.

#T7 - Application maintenance and support check

Description: The test checks if the user data is updated and checked and the user is given technical support whenever needed.

#T8 - Adaptability and scalability check

Description: The test checks if the application keeps track of multiple users and performs highly in the appropriate environment.

#T9 - User privacy check

Description: The test checks if the application protects the privacy of every one involved in the application and is safe against malwares.

#T10 - User audit check

Description: The test checks if the application documents the user data appropriately.

#T11 - User personalization and internationalization check

Description: The test checks if the application is suitable for people of different age groups, nationality and languages.

#T12 - User learnability and understandability check

Description: The test checks the ease of use of the application and how well the application can be interpreted

#T13 - documentation and training check

Description: The test checks if the application trains its users and documents everything for the users to explore and learn about the application.

#T14 - UI check

Description: The test checks if the application is visually pleasing and intuitive to the users.

#T15 - Application environment check

Description: The test checks if the application is correctly performing in the environment it is built and is correctly performing with its partner applications in the same environment.

#T16 - Application cultural and political environment check

Description: The test checks if the application is correctly performing in the cultural and political environment it is set in.

#T17 - Application legal environment check

Description: The test checks if the application is correctly bound by all its legal environments

#T18 - Application standard check

Description: The test checks if the team is regularly submitting working code, and following other Agile methodologies.

#T19 - Application latency check

Description: The test checks if the time to get the account information is less than three seconds.

III Design

22 Design Goals

#D1 - Latency Goal

Description: Designing a system where the client data retrieval from the Server takes less than 3 seconds.

#D2 - Scalability Goal

Description: Designing a system where multiple clients are able to access the application and use it smoothly.

#D3 - Privacy Goal

Description: Designing a system where the client data is available to only the client and it is protected from all the malwares.

#D4 - Real time Alarm Goal

Description: Designing a system where the user is alerted if the heart rate exceeds the acceptable range.

#D5 - Application Integration Goal

Description: Designing a system where the application can take the data obtained from partner applications such as Google Maps and integrate it with the system.

#D1 - Ease of use Goal

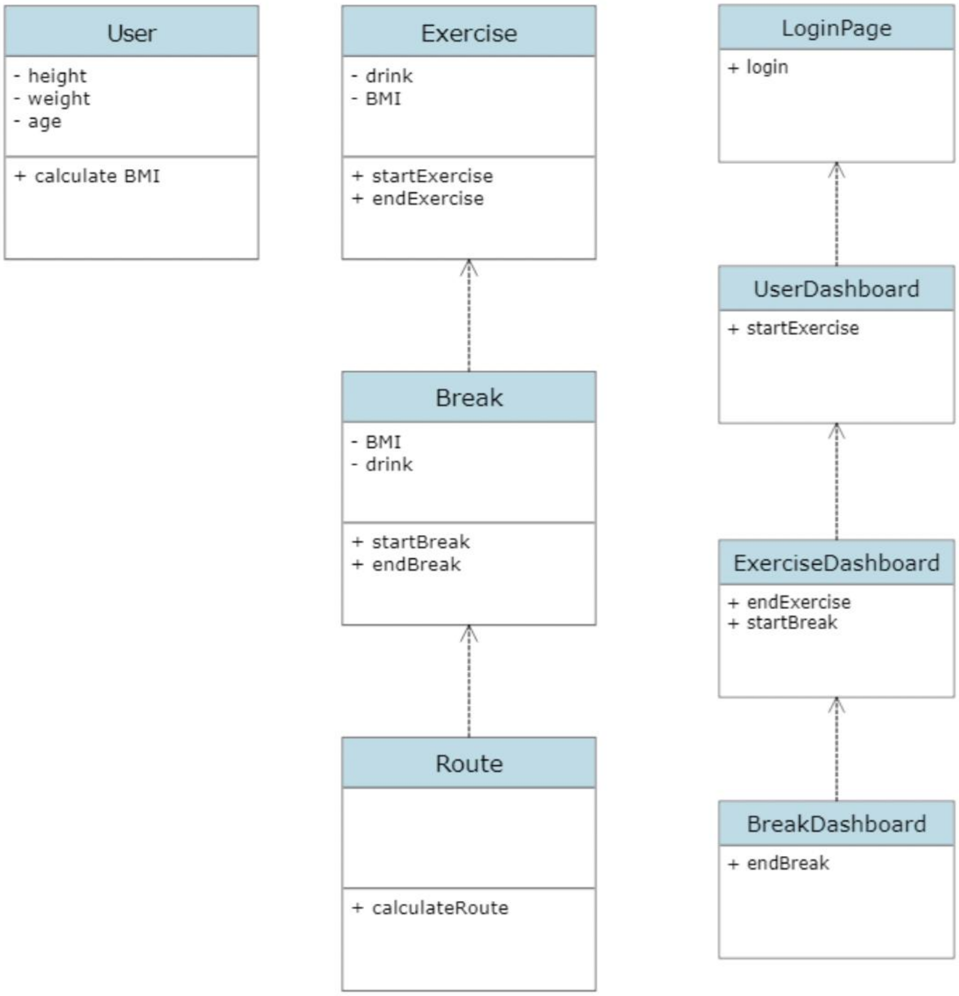
Description: Designing a system which is easy to use for all the age groups.

23 Current System Design

None

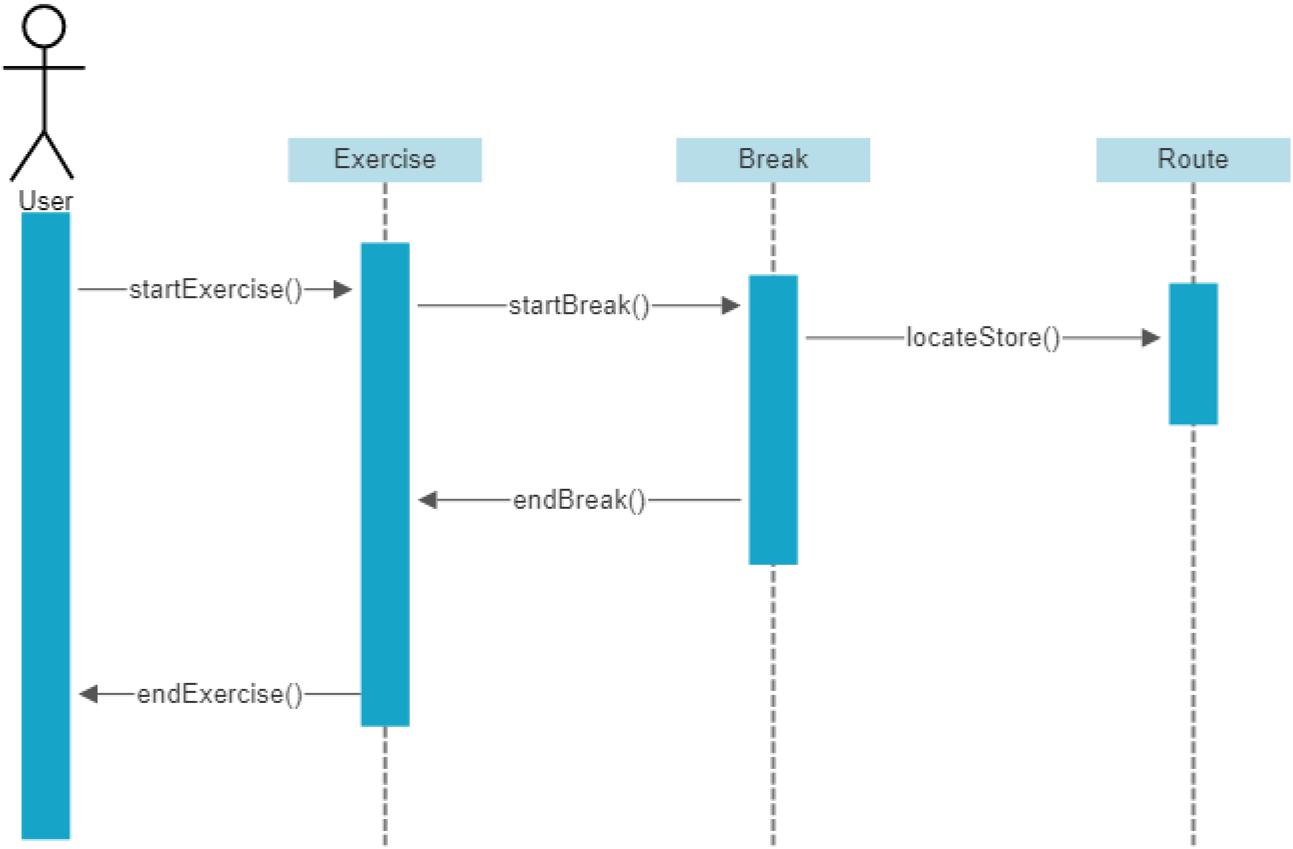
24 Proposed System Design

24a Initial System Analysis and Class Identification



24b Dynamic Modeling of Use-Cases

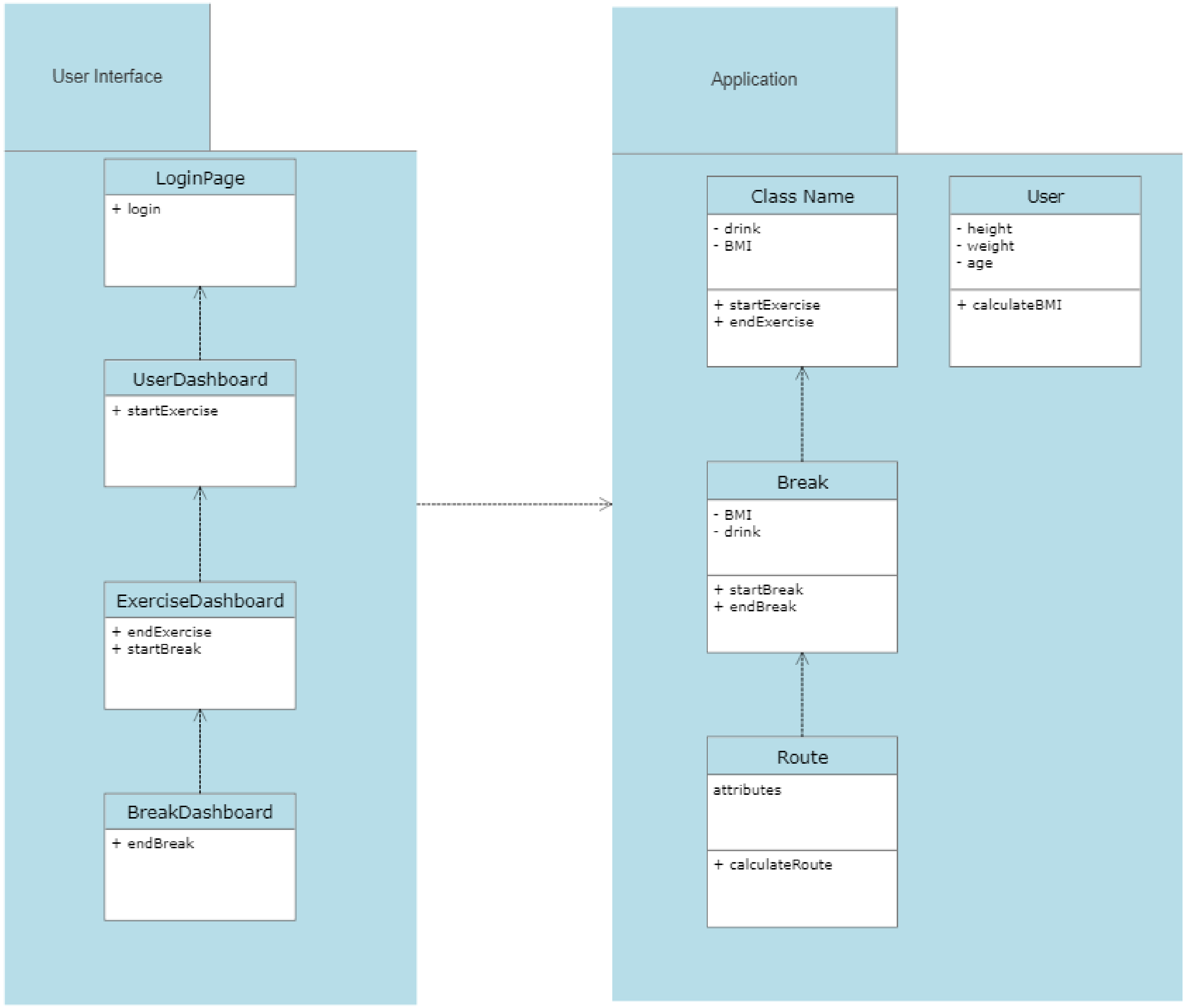
SV: Insert sequence diagrams of (at least the most important) use-cases, as a means of identifying other needed classes.



24c Proposed System Architecture

In this application, multiple users and a single server will be involved. We propose a system where clients are responsible for taking a user input, then sending over the input to the server, which is responsible for all the business logic and sending all the relevant information back to the client. The server will retrieve this information from the database. Therefore we think, Client Server Architecture is the appropriate choice here.

24d Initial Subsystem Decomposition



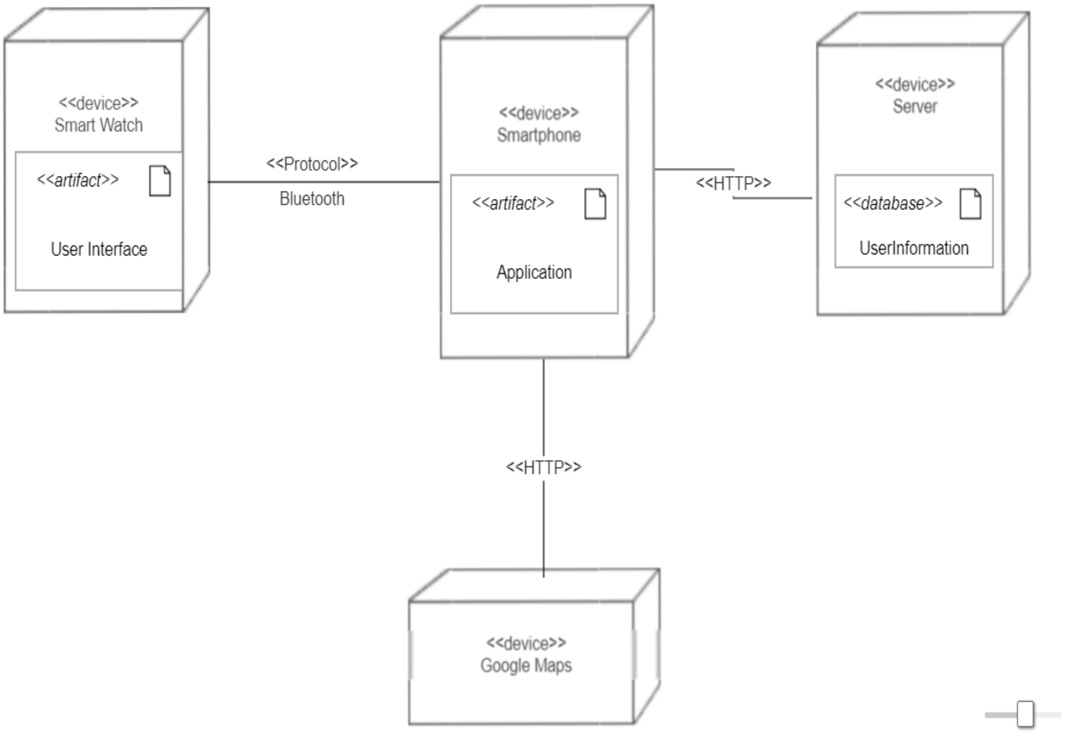
In this application, we have 2 subsystems. The first subsystem is the User Interface. It consists of LoginPage, which contains the login. Inside the UserDashboard we can start our exercise. Upon starting exercise, we have the option of ending the exercise or starting a break in ExerciseDashboard. If we start a break, we have the option of ending a break inside BreakDashboard. The second subsystem is Application. In this, we have the User details such as Height, Weight, Age. We calculate the Users BMI from this information. After the user starts an exercise and takes a break, we record the starting time of break, ending time of break, the BMI and drink from the Break Class. We also use the Route Class to check the attributes of the route the user has selected in order to calculate the distance of the selected route.

25 Additional Design Considerations

SV: The sections listed here do not need to be presented in the order given, and may not all be relevant for any particular project. Those that are relevant can help identify additional classes that are needed as a result.

25a Hardware / Software Mapping

SV: This is particularly important for distributed systems, such as those employing a client-server architecture. Use a deployment diagram to indicate which subsystems are mapped onto which piece(s) of hardware, and what communication subsystems need to be added to the system as a result.



The primary UI that takes care of users using the application and interacting with the application does so from the features available in the smartwatch. These features of the UI include prompt for the break, starting and ending break, prompt for locating a store and navigating a route to the store. The smartwatch is connected to the smartphone with a Bluetooth connection. The smartphone is where the calculation for the break is done. The smartphone however is connected to both a server and google map application. The server holds the User Information stored on the server which is relayed back and forth to the application on the smartphone, whereas the Google Maps is a third party application that is used for navigating a route to local nearby stores. Both Google Maps and server are connected through HTTP for proper encryption and secure data transfer of necessary information and functionality.

25b Persistent Data Management

SV: Document the classes and perhaps subsystems necessary to store persistent data when the system shuts down, and to restore that data when the system starts back up again.

*Reiterate key data structures and information as necessary for the understanding of this design phase. Refer the reader back to the data dictionary in section **Error! Reference source not found.** to avoid undue repetition, while reviewing only the most relevant items here.*

The application on the smartphone has a class that stores the User Information like height, weight, age, type of exercise, although this information is relayed to the server for being stored, the only way the server affects the application is in terms of logging in and signing up. In case of any server outage or the systems that are online shuts down, it does not affect the application as all the required information for the computation of the break is done locally on the user's machine. When the servers restart, the connection with the device is reestablished and the data for signing up and logging in can be accessed again. The data that is in the server is stored in a physical hard-drive so even if the system shuts down, there is no data loss.

25c Access Control and Security

SV: Identify the access control and security concerns for this system, and the new classes and/or subsystems that must be added to handle those concerns.

When it comes to User Information, this data is protected and only accessible to the user the data belongs to, even the administrators and people working with the server do not have access to this data. The reason this is set in this way is so that we can protect our users' privacy. The User Information is stored with a specific user id (UUID, which is randomly generated and assigned) this information is stored in a hash table and allocated according to the UUID's.

25d Global Software Control

SV: Identify the global software control concerns for this system, and the new classes and/or subsystems that must be added to handle those concerns.

There could be an issue where a new user tries to create an account on their phone while they are not in a place to connect to the server. This would mean that the user's information will not be stored in the server. To fix this, the user class will be able to store the user's data in the application to allow the user to still use the product, and when it can connect to the server, it will ask the user if it allows the application to send the user information to the server. If allowed it will send the data.

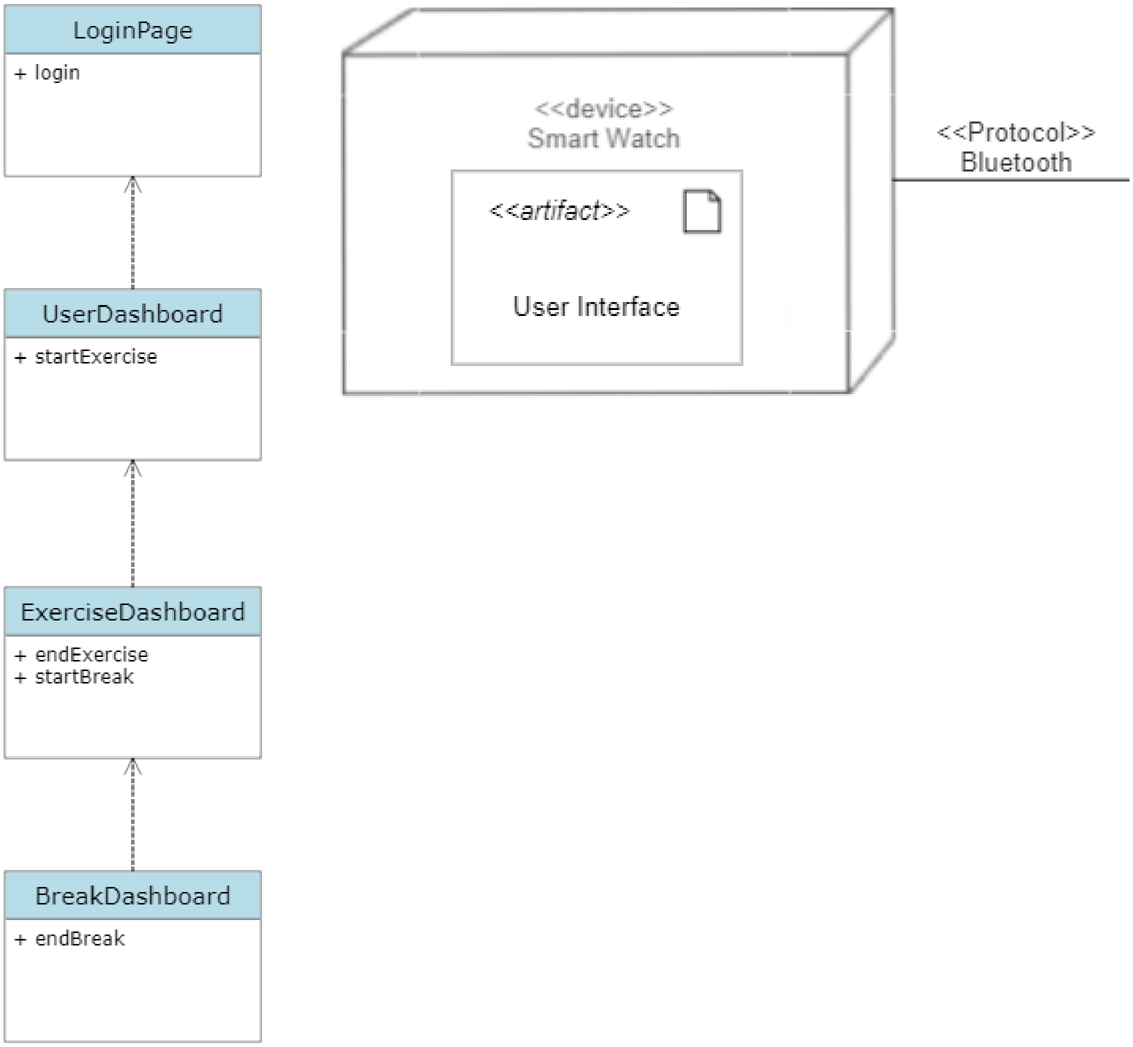
25e Boundary Conditions

SV: Identify the boundary condition concerns for this system, and the new classes and/or subsystems that must be added to handle those concerns. In particular consider startup, shutdown (normal or abnormal), and the creation and/or maintenance of any configuration files, databases, or similar supporting data files.

When the system starts up, all the data that is held in local devices is sent to the database on the server to either store in new data or update the data that was stored before. During maintenance the connection between the local device and servers is cut off, this does not affect the functionality of the application, but the database however stores all the data locally on a separate storage device so that no data is lost during the maintenance.

25f User Interface

SV: Include a preliminary user interface design here, possibly as a rough sketch or *other mockup, in order to identify additional classes needed to implement the interface.*



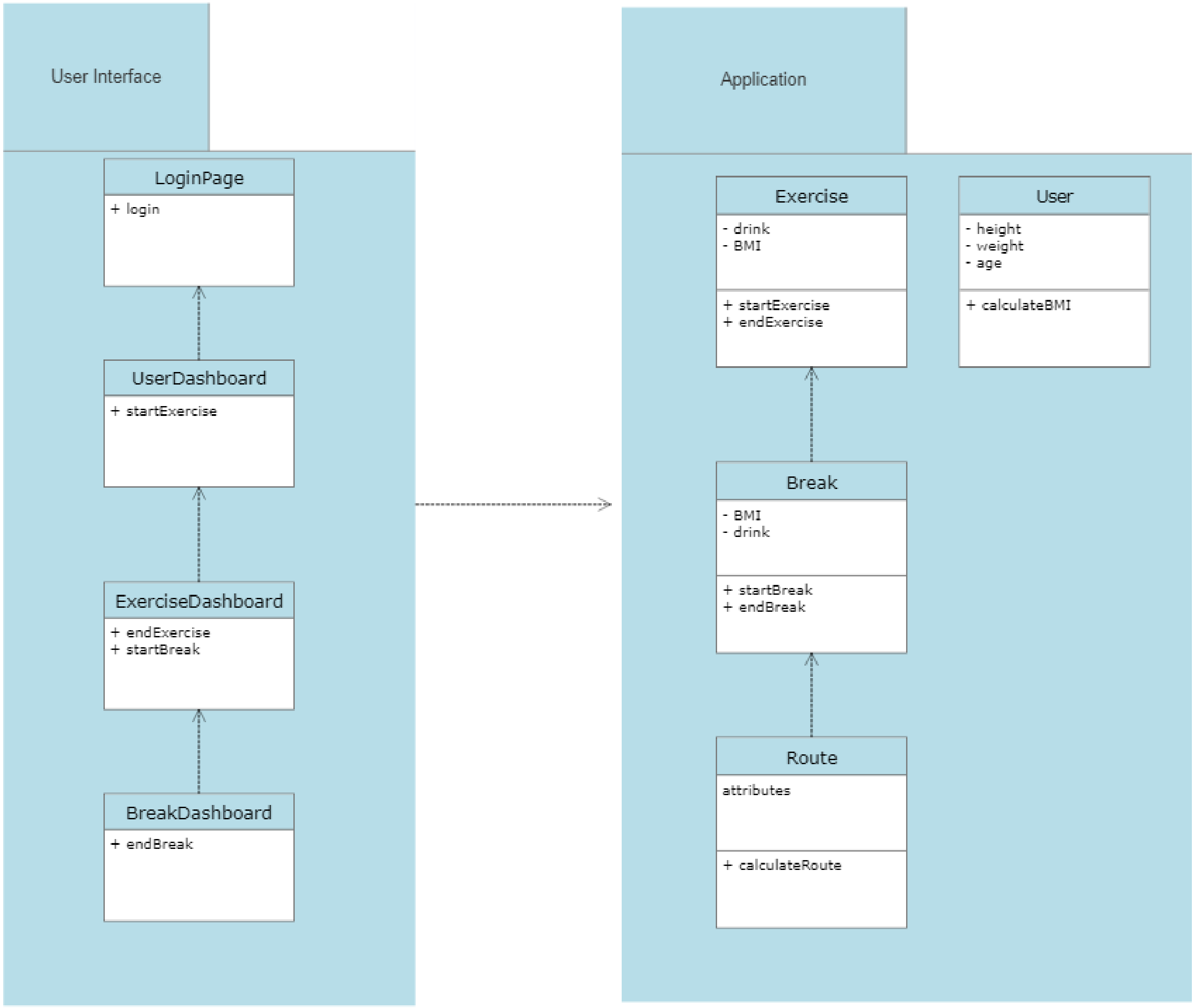
25g Application of Design Patterns

SV: Any design patterns applied as a result of previous sections should have been addressed there, and identified as such at the time. Use this section to document only the additional design patterns that were not previously covered elsewhere. (If any.)

There are no additional design patterns used in this product at this time.

26 Final System Design

SV: Include here the final version of the overall system design, incorporating all the subsystems and classes added as a result of additional design considerations. Multiple *diagrams may be needed, possibly starting with an overall package diagram showing all the different subsystems and the (important) classes contained within each one. Still not a lot of internal details.*



27 Object Design

This section documents the internal details of each class, to the extent that they can be designed at this time. Included should be the class interfaces (public method signatures and responsibilities) and constraints. It is probably best to break this section up into subsections corresponding to subsystems as documented above, and/or by (Java) packages if those are designed. It may also be appropriate to address additional design pattern considerations here, but not to the point of being redundant of previous *documentation.*

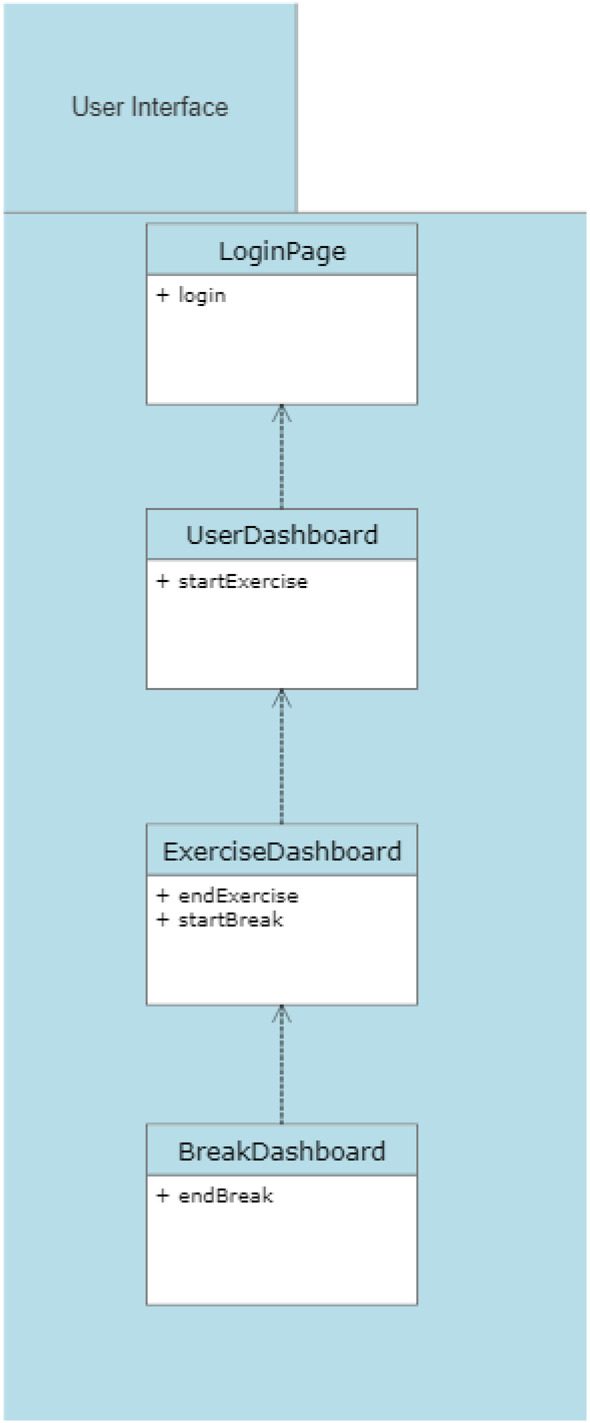
Certain methods, such as simple getters, setters, and constructors are not always documented, unless there is something special about them such as in the Singleton or Factory Method design patterns.

27a Packages

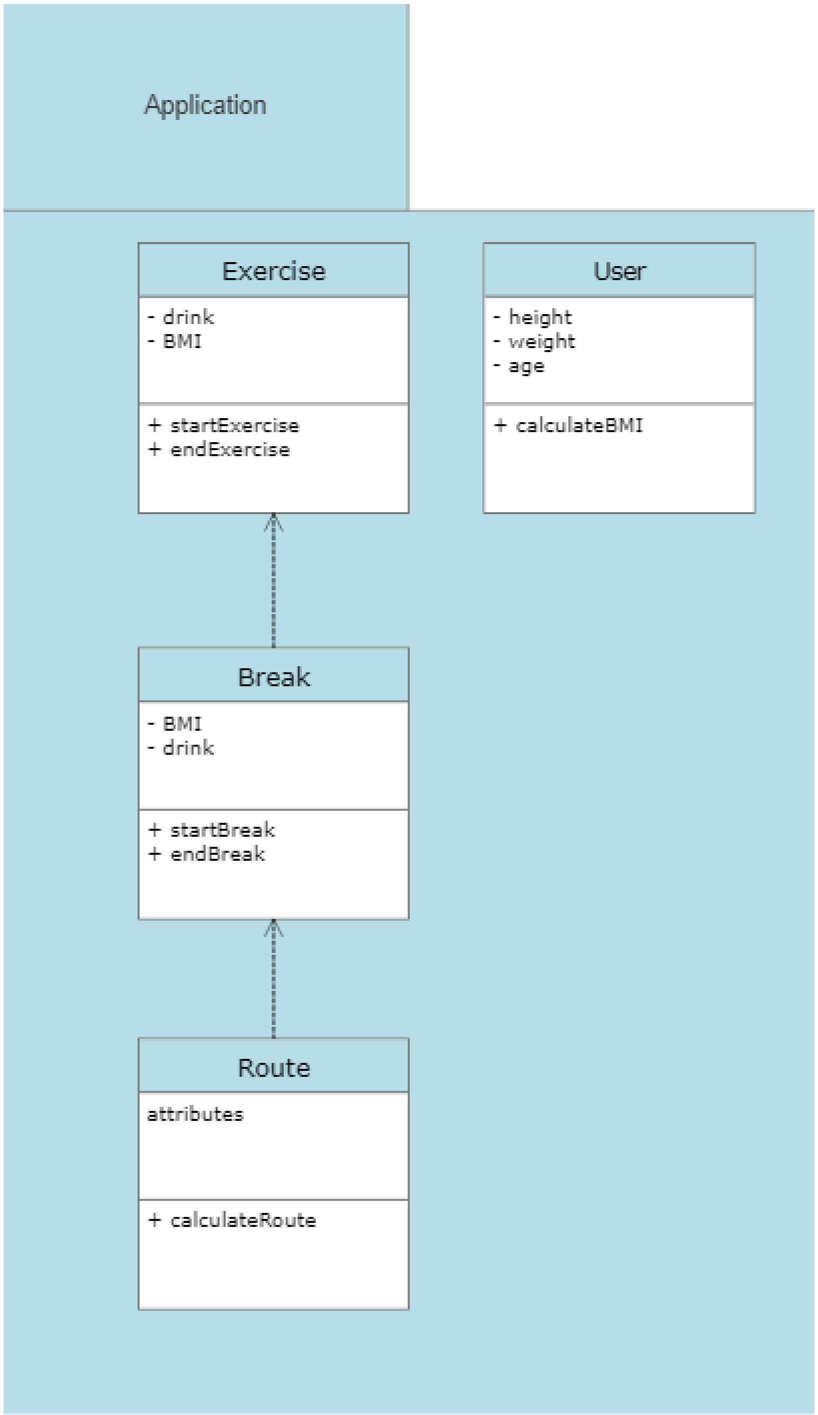
SV: If the design involves assigning classes to packages (.e.g Java packages), then the packages to be created should be documented here.

As of the moment, there are no packages that need to be used with these classes.

27b Subsystem I



27c Subsystem II



IV Project Issues

28 Open Issues

SV: Issues that have been raised and do not yet have a conclusion.

An issue that has been raised is if the user is in an area that doesn't have any stores nearby. This would render the store finding aspect of our product pointless as any store would be too far away for the user to actually go there.

29 Off-the-Shelf Solutions

SV: Discussion of products or components currently available that could either be incorporated into the new solution or simply used instead of developing (parts of) the new solution. The distinction between sections 35 a, b, and c is subtle, and not very important.

29a Ready-Made Products

SV: Products available for purchase that could be used either as part of a solution or instead of (a part of) a solution.

The user will need to have a smartwatch that can track the user's heart rate. We will not create our own smart watch as that is beyond the scope of this project but it is a necessary product to have so we can track the users heart rate and create a proper application for it.

29b Reusable Components

SV: Similar to 35a, but for components such as libraries or toolkits instead of fully blown products.

We decided to use Google Maps instead of creating our own navigation system. We decided on doing this because we know that Google Maps is a very reliable product that does the task we need it to. Creating our own navigation and map system would drastically increase the development time and could be very easily prone to errors.

29c Products That Can Be Copied

SV: Products that could legally be copied would typically be past projects developed by the same development group, provided there were no restrictions that would prevent their reuse.

There are no products on the market that respond to the needs fulfilled by this application, which makes it difficult to assume that the product could be copied. There are applications that target features like hydration status, and oxygen levels which if were to be implemented in our application would be quality of life updates for the application.

30 New Problems

SV: The proposed new system certainly has its benefits, but it could also raise new problems. It is a good idea to identify any such potential problems early on, rather than being surprised by them later.

30a Effects on the Current Environment

SV: Could the new system have any adverse effects on the working environment, e.g.

the way people do their jobs?

The application is designed to work with a server that has User Information and this information is stored locally on a storage device, but in future if the storage procedure were to change from storing data locally to storing it online, for example cloud storage then that changes the team responsible for the server maintenance. This could mean a change in teams dedicated to the procedure of maintaining the servers and a different skill set for the team.

30b Effects on the Installed Systems

SV: Could the new system have any adverse effects on other hardware or software systems?

If the system switched to using cloud servers, the application would need to be able to connect to this new server which means the application would have to be partially rewritten to account for this change. The data would also need to be transferred from the physical servers to the cloud servers.

30c Potential User Problems

SV: Could the new system have any adverse effects on the users of the software? Could users possibly have a negative response to the new system?

The user would need to update their application on their smart device so the new code for the application to run on a cloud server could be implemented. If they don't and try to connect to the application, the system might not be able to connect to the server and the user might not be able to get their information.

30d Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

SV: Are there any (physical) limitations in the expected environment that could inhibit the proposed product? (e.g. weather, electrical interference, radiation, lack of reliable power, etc.)

If the user is hiking/camping in the woods or a hilly area, she/he will not be able to find stores for hydration nearby. The application will tell you the location of the stores but it will be too far from that place (as it is isolated). So the user would need to find a stream to rehydrate which is the physical limitation.

30e Follow-Up Problems

SV: Basically any other possible problems that could occur.

There is a possible problem in regards to the navigation feature of the application. If Google Maps stop working then that affects our applications ability to locate a store entirely. As the application does not have an inbuilt navigation system made for locating stores and relies heavily on Google Maps, this means any problem with Google Maps would affect the applications functionality as well.

31 Migration to the New Product

Not Applicable

32 Risks

#R1 - Server overload

Description: If the server is bombarded with a lot of client requests more than it can handle, the client will have latency issues or worse, the server might crash.

#R2 - Highly Sophisticated Malware

Description: If a highly sophisticated malware which has not been encountered by the privacy team is used, there is a chance it could penetrate the system and steal the private data.

#R3 - Unreliable Internet Connection

Description: The application relies on the user to have a reliable internet connection, if the user has issues with the internet, the application will not be able to receive the users data and recommend breaks to the user. The application will not be able to load all the potential stores where users can have a drink.

#R4 - Inaccurate Heart Monitor

Description: The application relies on the user to have a watch which has an accurately performing heart monitor. The application will not perform ideally and might malfunction if the heart monitor is inaccurate.

#R4 - Inaccurate User Information

Description: The application relies on the user to update the information such as Height, Weight, Age. WIthout this, the application might not be able to accurately give guidance to the user.

33 Costs

We estimate that this project will take around a year to complete. We will need to hire around 5 Software Engineers, 1-2 Fitness Professionals, 3 person recruiting team, assuming the application will be made available in 10 languages in the first release, we would need 10 more Translation specialists on a temporary basis. We would need a reliable server, preferably a cloud based like AWS, this would increase the reliability of the server. We would also need a database to store all the user information. We're thinking about using Amazon S3 for Database. We would also need an office space where all the hired people can interact with each other. We would also require them to provide them with additional perks to better retain the employees. In case, an employee decides to leave before the project completion, we would also require recruiters to look for an appropriate candidate again. All things considered, we estimate the total cost of the Project to be 2 Million Dollars.

34 Waiting Room

1. The application could be extended to all the major languages used in the world.
2. The application, if successful, can design its own brand of watch, which can better integrate with the application than other smartwatches.
3. The application could be extended to users forming groups and discussing going for runs/hikes together.
4. The application could be extended to hikers, where the hikers can know the location of other hikers doing the same trail. This feature would only be applicable to the users who have given the permission to display their location to other hikers.

35 Ideas for Solutions

We believe that since this application is primarily going to be a mobile application, the developers would need to use Android Studio for making android application and Swift for iOS. We recommend using InTelliJ IDEA as the IDE for Android Studio and XCode for iOS. We recommend deploying the code on the AWS Server, and using Amazon S3 for creating the database. We also recommend using a relational database management system such as MySQL since the data which is currently being stored in the database is heavily structured.

36 Project Retrospective

We were successfully able to create a design of a potential project which can be used in the real world. The goals set by the project are realistic and can be implemented in relatively short periods of time. We were able to work as a group and come up with different ideas about what the features inside the project should look like. Everybody contributed from their end. Since most of the projects rely on having a Client and Server, we believe that forming groups based on skill (eg. having 2 group members with front end skills and 2 with backend skills) is useful. We were able to meet in person for weekly meetings and had quick chats on Discord. Overall, we believe that the process worked really well.

V Glossary

SV: The glossary is a more complete and inclusive dictionary of defined terms than that found in section I.7.a, the latter of which only covered the most important key terms needed to understand the report.

Your text goes here . . .

VI References / Bibliography

This section describes the documents and other sources from which information was gathered. This sample bibliography was generated using the "Insert Citation" and "Bibliography" buttons in the "Citations & Bibliography" section under the "References" tab of MS Word. Creating new citations will not update this list unless you click on it and select "Update Field". You may need to reset the style for this paragraph to "normal" after updating.

[1] Robertson and Robertson, Mastering the Requirements Process.

[2] A. Silberschatz, P. B. Galvin and G. Gagne, Operating System Concepts, Ninth ed., Wiley, 2013.

[3] J. Bell, "Underwater Archaeological Survey Report Template: A Sample Document for Generating Consistent Professional Reports," Underwater Archaeological Society of Chicago, Chicago, 2012.

[4] M. Fowler, UML Distilled, Third Edition, Boston: Pearson Education, 2004.

VII Index

This section provides an index to the report. The sample below was generated using the "Mark Entry" and "Insert Index" items from the "Index" section on the "References" tab, and can be automatically updated by right clicking on the table below and selecting "Update Field". To remove marked entries from the document, toggle the display of hidden paragraph marks (the paragraph button on the "Home" tab), and remove the tags shown with XE in { curly braces. }