Music-driven workout app proposal

Manasvi Sagarkar and Lucy Newman

January 20, 2018

1 Overview

We propose to build a workout app that customizes workouts for users based on their music taste. The key components of the proposed app are as follows:

I. The minimum viable product for this project will recommend pairings of song and exercise based on the user's goals (type of workout, intensity, etc.) and choice of song or genre. We will leverage the Spotify API and a database of exercises to find a pairing of song and exercises such that the exercise can be done to the beat of the song.

II. We will add several features to motivate users to continue using the app. Users will have levels which can be promoted by doing well in tests and completing their goals.

   1. A Level 1 user can log in, enter goals and motivation information, test themselves, work out, or form a team of mutually supportive friends to hold each other accountable for completing goals.
   2. At Level 2, a user is able to choose and complete a Theme, where their workouts for a chosen duration are seeded by an artist or genre of their choice.
   3. At Level 3, a user can join a competition.
   4. At Level 4, a user can start a competition.
   5. At Level 5, a user can add their personal exercises to the general database.

The ideal version of this program will include an Apple Watch app, but depending on the skills and interests of our team members, we will instead make all of the same features available via a different user interface and slightly different actions on the iPhone, for the purpose of this class. When an action is performed differently for the Apple Watch version compared to the iPhone version, it will be designated in this document by a box, as below.

Including an app for the Apple Watch could allow some additional features and improve the user experience in a number of ways, for example:

- Fitness tests can use motion data from the Apple Watch to determine the number of reps and the rate at which the user completes the exercise rather than require the user to count and enter repetitions themself.
- We can allow users to add an exercise by beginning the exercise while wearing the watch. As they are completing the exercise, the app can dynamically determine the rate at which they are completing the exercise and begin playing a song as they are doing it.
- We can track health measures such as heart rate, reward users who are pushing themselves and adjust workouts based on the data.

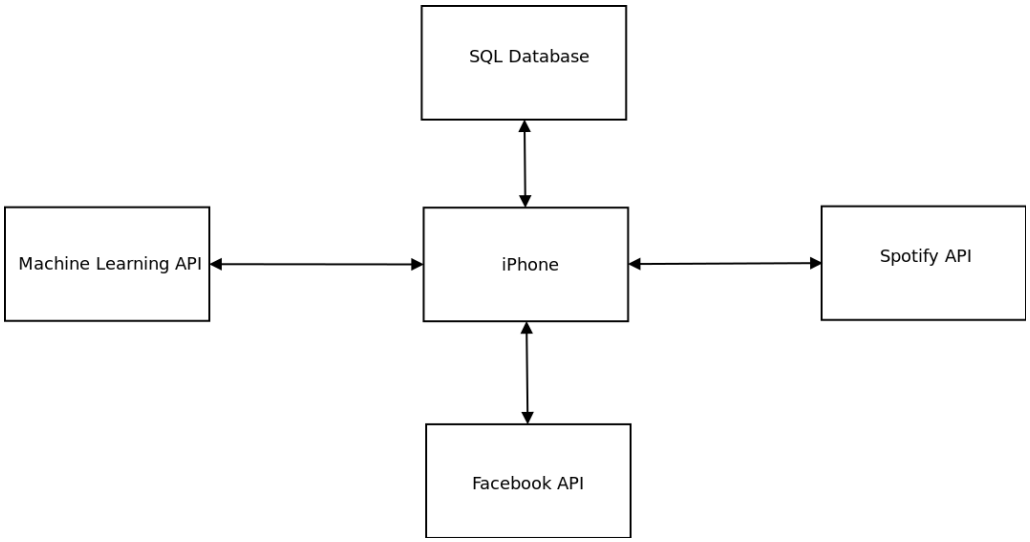
We will describe both alternatives wherever an Apple Watch app would lead to different actions.

2 Hardware, Programming Language & IDE

We have two possible scenarios for how computation will be distributed. We will include information flow diagrams and use case diagrams for both options in this proposal, and determine which version is appropriate based on the experience and goals of the larger group. Both scenarios involve utilizing the Spotify API and a machine learning library for recommending songs, the **wger Workwout Manager API** for recommending workouts, and the **Facebook API** for logging in, finding friends, and external sharing. We will use a **SQL Database** for storing and retrieving a user's information.

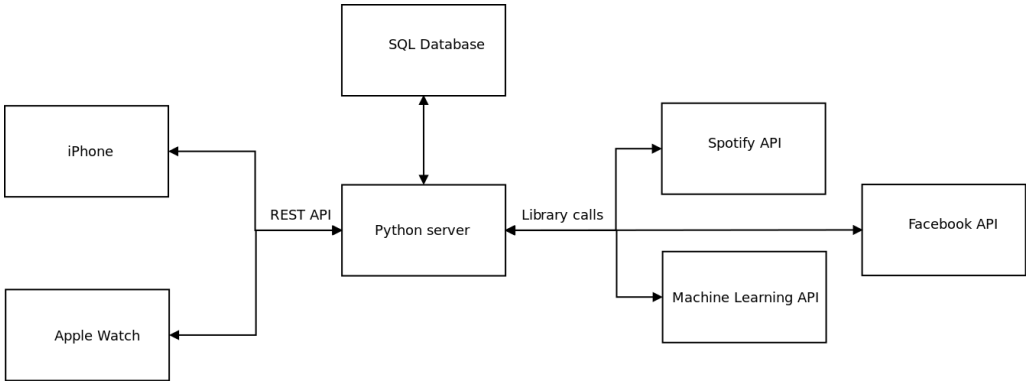
The simpler architecture design will use **C++** for the majority of computations, and will make these computations locally on the iPhone.

Case 1: Information Flow Diagram for App Run Locally on an iPhone



In the alternate version, the back end of the app is coded in **Python** and hosted on a web server. **C++** is still used for interacting with app interfaces and making requests to the REST API. Advantages of this strategy include making the code more portable between different devices, and that we both have more experience with Python than with C++. A diagram of the information flow is included below.

Case 2: Information Flow Diagram for App Run on a Server



If we include an **Apple Watch** app written for Apple's **WatchOS**, then the second strategy would be preferable, since it would facilitate communication between devices and avoid repeated code.

3 Prioritizing Features

3.1 High priority

The minimum viable product for this app includes:

- Authentication
- Fitness testing
- Working out

3.2 Medium priority

Additional features, in rough order of priority for implementation. If the group feels that the scope of the project is too large, then one or some of these items can be cut. Which ones are cut can be decided by the larger group:

- Set goals, give motivation and fitness information
- Choosing themes
- Competitions
- Teams

3.3 Low priority

Features but would be nice for an optimal version of this product, but likely will not be focused on for the purpose of this class

- Making personal custom exercises public

3.4 Apple Watch features

The Apple Watch app will either be high priority or low priority depending on the direction the group decides to go with this project.

- Using motion sensing data to determine period of exercise and count reps
- Using heart rate data to reward user
- Using heart rate and motion sensing data to determine whether a user is able to stick to the beat of a song, and adjust intensity accordingly

4 Why This App?

4.1 Motivation

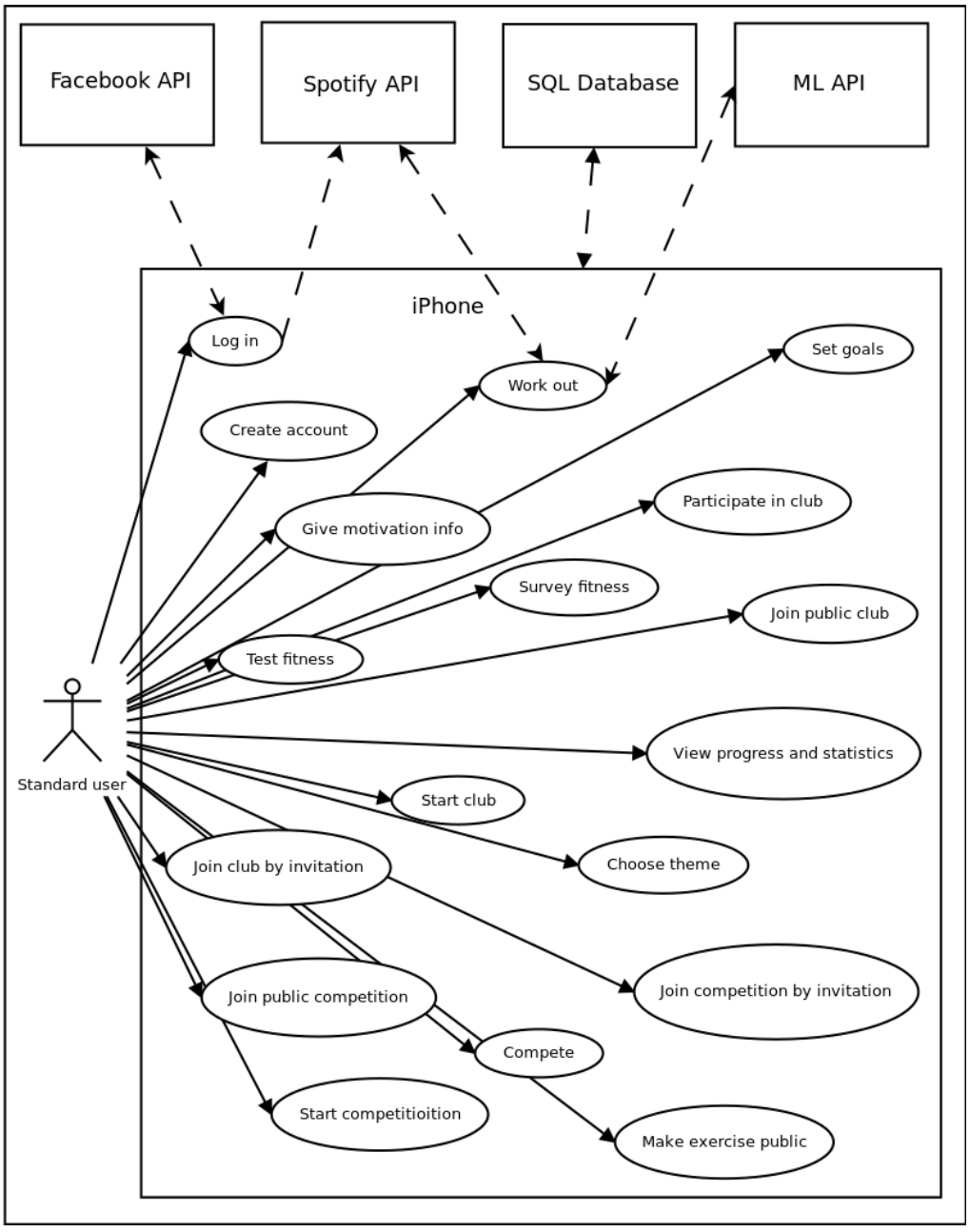
We both find that music can be a great motivator, and we both enjoy working out and feel that having good music with which to work out to can be a huge factor in how well we are able to stick to a workout. We came up with this idea because it is something that we would want to use for ourselves, but it would be a great product for a wide range of users, from a someone casually wanting to get back into shape to a dedicated figure skater who wants to find a song to match a step sequence. In our research we didn't see any products available quite like ours, that specifically matches the workout to the music, so we think it is something that should be done! Everyone loves music, and working out to music can be a really fun experience, especially combined with the social and competitive motivation we plan to add.

4.2 Why This Proposal Should Be Accepted

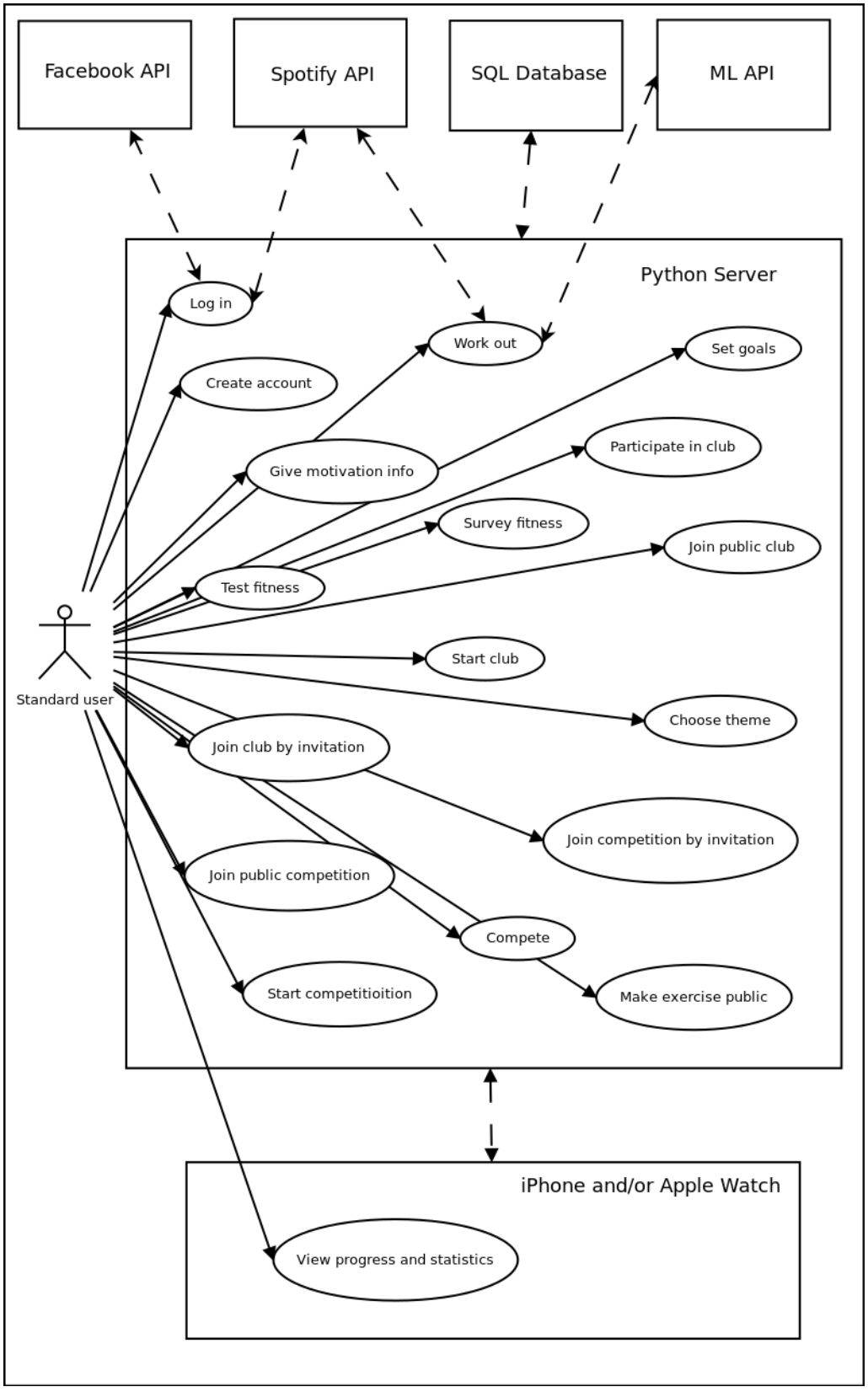
In addition to being a useful app, this would be a good proposal for this class because it requires a wide range of skills, and can take advantage of the various backgrounds of people in this class, from UI design and implementation, to working with databases, to machine learning. Additionally, since many features are self-contained, this project can easily be adjusted in scope to fit the constraints of the class.

5 Functional Requirements

Case 1: Use-Case Diagram for App Run Locally on an iPhone



Case 2: Use-Case Diagram for App Run on a Server



5.1 Create Account

Main Scenario

Step 1: The user is given options of logging in through Spotify creating a new account.

Step 2: If the user chooses to use Spotify, they will be prompted to log in via Spotify.

Step 3: If successful, an account is made for the user, and they are redirected to the Home Page.

Extenstions

Step 2a: If the user chooses to create a new account directly, they will be prompted to enter login information.

Step 3a: Account creation is unsuccessful, the user will be informed of this, and returned to step 2 or 2a.

5.2 Log In

Main Scenario

Step 1: The user is prompted to enter their login information.

Step 2: If the login information is successful, they are redirected to the Home Page.

Extenstions

Step 2a: If the login is unsuccessful, the user will be informed of this and returned to step 1.

5.3 Give Motivation Information

Main Scenario

Step 1: Ask the user if they would like to fill out the survey.

Step 2: If yes, they answer questions such as:

- What music they want to work out to. If they logged in through Spotify, their music data will be used to make personalized recommendations.
- Whether they are motivated by praise or competition.
- What words and phrases out of a given set they find the most motivating.

Step 3: Thank/congratulate user for completing the survey and redirect to the Home Page.

Extenstions

Step 1a: If no, redirect the user to the Home Page.

Step 2a: If the user quits the survey in the middle of filling it out, the app will save their responses and open the survey to that point if they fill it out later. It will also use the responses they filled out.

5.4 Survey Fitness

Main Scenario

Step 1: Ask the user if they would like to fill out fitness survey.

Step 2: If yes, they answer questions such as:

- Height, weight and age.
- How many hours a week they currently exercise.
- What their current routine is if they have one.
- Whether they play any sports and at what level.
- What their current benchmarks are for different exercises such as running, weights, core exercise etc.
- The exercise equipment they have access to.

Step 3: If the user doesn't want to fill out the survey, they can go straight to the next step in setting up which setting their fitness goals.

Extenstions

Step 2a: If the user quits the survey in the middle of filling it out, the app will save their responses and open the survey to that point if they fill it out later. It will also use the responses they filled out.

5.5 Test Fitness

Main Scenario

Step 1: The user is prompted to choose a testing category (cardio, general strength, arms, abs, etc.).

Step 2: The user is prompted to select exercises they want to test, from a set of exercises, some of which they have already tested on, and one which they haven't.

Step 3: The user is shown a timer and told to count how many repetitions they complete in 30 seconds.

Step 4: The user is asked to record the number of repetitions.

Step 5: The user is returned to step 3 for each of the exercises they are testing on.

Extenstions

Step 2a: If the user has never tested in this category before, all exercises are new.

Step 3a,4a: If the user quits in the middle of the test, the current information is saved.

Step 5a: If there are no exercises left to test on, the user is congratulated for completing the test.

Apple Watch version:
Step 3: The user selects a button to begin the test.
Step 4: The apple watch analyzes the frequency of movement to determine the number of repetitons.

5.6 Set Goals

Main Scenario

Step 1: The selects a general workout category from several options (cardio, abs, etc).

Step 2: The user selects an intensity.

Step 3: The user selects a number of days per week.

Step 4: The user selects days they want to work out and times on those days.

Step 5: The user is thanked/congratulated for completing the survey.

Step 6: The user is redirected to the home page.

Extenstions

- The user can skip any step. If step 3 is skipped, then step 4 is skipped automatically.
- The user can quit at any step. If the user quits, the current responses are saved.

5.7 Work Out

Main Scenario

Step 1: The user selects "Begin Workout."

Step 2: The user is asked whether they want to specify additional data for a particular workout.

Step 3: If yes, they are asked what workout category and intensity they want.

Step 4: If the user is not in a Themed Workout, a seed is chosen based on the user's music taste data, if available.

Step 5: An exercise is chosen from the category that the user has selected, or, if the user has not chosen a category, then at random.

Step 6: Song-workout pairs are chosen based on the optimal matching between the user's workout goals and the music seed, and these are displayed to the user.

Step 7: The user is congratulated and given points for completing the workout.

Extenstions

Step 3a: If no, move on to step 4.

Step 4a: If the user is in a Themed Workout, the seed is already set based on the theme.

Step 4b: If the user does not have music taste data available, a seed is chosen from a selection of Spotify workout playlists.

Apple Watch Version:
Step 6a: If the user's target heart rate data is available, the user is given Fire for maintaining their target heart rate.

5.8 Start Club

Main Scenario

Step 1: The user chooses a name for the club and provides a short description of its purpose.

Step 2: The user decides whether the club is private or public.

Step 3: If the club is public, the club is published and all users can view its name and the description of its goals.

Step 4: The user is prompted to invite people to the club.

Step 5: The user is welcomed to the club, redirected to the Club Page, and encouraged to customize the club page.

Extenstions

Step 3a: If the club is private, it is not published.

Step 4a: If the club is public, the user can skip this step.

5.9 Join Club by Invitation

Main Scenario

Step 1: User accepts invitation.

Step 2: User is redirected to Club Page.

Extenstions

Step 1a: If the user declines an invitation, it is deleted.

5.10 Join Public Club

Main Scenario

Step 1: User visits Club Listing page.

Step 2: User searches for interests.

Step 3: If the user finds a club that they want to join, they selects "Join."

Step 4: The user is added to the club and redirected to the Club Page. The club is added to the user's list of clubs in the My Clubs page.

Extenstions

Step 1a: The Club Listing page displays suggestions for the user based on their workouts and preferences so far. They may select a club from these suggestions and skip to step 3.

Step 3a: If the user doesn't find a club they want to join, they are returned to the Club Listing page.

Step 4a: If the user has already been removed from this club, the user is reminded of this, and is not added to the club.

5.11 Participate in Club

Main Scenario

Step 1: The user selects the club from their My Clubs page

Step 2: The user is redirected to the Club Page for this club.

Step 3: The user posts to the club page.

Extenstions

Step 3a: The user likes an existing post.

Step 3b: The user comments on an existing post.

Step 3c: The selects "Leave Club." The club is removed from the user's list of clubs and the user is redirected to their Home Page.

5.12 View Progress and Statistics

Main Scenario

Step 1: Basic info is displayed on the Home Page, and the user can click on this to see more detailed info.

Step 2: From the View Progress page, the user can navigate between their personal statistics and those of any teams of which the user is a member, as well as competitions they are a part of.

5.13 Choose theme

Main Scenario

Step 1: From the Home Page, the user selects Choose Theme.

Step 2: If the user's music preference data is available, the app will suggest theme artists based on this data.

Step 3: The user can search an artist, song, or genre.

Step 4: The user can select an artist, song, or genre.

Step 5: The user is asked to select the duration for which this will be their musical theme.

Step 6: The user's musical theme is updated to this artist, song, or genre.

Extenstions

Step 1a: If the user does not have sufficient privileges, the action will fail. The user will be told what they have to do to get these privileges, and will then be redirected to the Home Page.

Step 2a: If music preference data is not available, the app will make suggestions based on Spotify workout playlists.

Step 3a: The user can make a selection based on the given suggestions, without searching. Move to step 5.

Step 4a: If the user is unsatisfied with these results, they can search again. Move to step 3.

5.14 Start Competition

Main Scenario

Step 1: The user chooses a name for the competition and provides a short description of its purpose.

Step 2: The user selects the exercises to be a part of the competition.

Step 3: The user chooses the duration of the competition.

Step 4: The user decides whether users can make multiple submissions.

Step 5: The user decides whether the competition is private or public.

Step 6: If the competition is public, it is published and all users can view its name and the description.

Step 7: The user is prompted to invite people to the competition.

Step 8: The user is redirected to the Competition Page.

Extenstions

Step 1: Step 1a: If the user does not have sufficient privileges, the action will fail. The user will be told what they have to do to get these privileges, and will then be redirected to the Home Page.

Step 6a: If the club is private, it is not published.

Step 7a: If the club is public, the user can skip this step.

5.15 Join Competition by Invitation, Join Public Competition

The logic here is nearly identical to that for joining groups. The only difference, beyond substituting "Club Page" for "Competition Page," etc., are:

- If the user does not have sufficient privileges to join a public competition, the action will fail. The user will be told what they have to do to get these privileges, and will then be redirected to the Home Page.
- No one can be removed from a competition, so it is not necessary to check whether they have been removed before allowing a user to join a competition.

5.16 Compete

Main Scenario

Step 1: User selects "Begin Competition."

Step 2: User is tested on the competition exercises (see Fitness test).

Step 3: User is redirected to the competition results page.

Step 4: If the competition does not have a limit on the number of entries, or the user has not exceeded the limit, they will have the option of completing the competition again. If they choose this option, return to step 2.

5.17 Make Exercise Public

Main Scenario

Step 1: The user selects an exercise to be made public.

Step 2: The exercise user is prompted to enter descriptions and add a video of themselves completing the exercise.

Step 3: The exercise is added to the database.

Extenstions

Step 1a: If the user does not have sufficient privileges, the action will fail. The user will be told what they have to do to get these privileges, and will then be redirected to the Home Page.

Step 3a: If there is another exercise in the database which the app suspects is the same as the one attempting to be added, the action will fail. The user will be asked whether they want to add a different exercise. If yes, go to step 2. If no, redirect to Home Page.