HIITMix

Design Document Term Project

Billy Dixon
11/20/12

Fall 2012
CSCI498A Mobile Device Programming
Colorado School of Mines

Overview

This application, HIITMix, is an interval workout timer that creates custom playlists from the user's library to match the intensity of the workout. Using a track analysis API from EchoNest, the app will create a database with detailed information on the tempo, energy, volume, and other necessary metadata of each song in the user's library. After the user selects an interval workout, either pre-made or custom, a playlist will be created that will match a song's tempo to each interval's target heart rate as closely as possible.

The idea behind this is that there is a correlation between heart rate and the tempo/energy level of the song a person is listening to. Rather than starting from the beginning of each song, this app will select the best section of the song to match the workout interval in length and intensity. Users will have complete control over the creation and editing of both workouts and playlists if they don't like the selection that HIITMix picks. Using text-to-speech to alert the user how much time is left or what intensity to go for is a feature I would like to implement if time permits.

User Stories

Jeff McBroFist is at the gym trying out this new high intensity interval training everyone is raving about. He turns on the radio hoping to find a station playing some decent music to get him pumped up. But of course, they're all playing either Justin Beiber or commercials. In a fit of rage over hearing 5 Beiber songs in a row, he throws the radio out the window and picks up his trusty smartphone and hits the shuffle button. A decent song comes on and he start getting pumped. His heart rate climbs, he's jamming out to the song, and then... the interval ends. He has to stop and rest for a minute. But the song is still going. And he loves this song. He doesn't want to calm down. He wants to keep going. But that's not how the training method works. So he hits skip and a nice slow song comes on. His heart rate drops as he gets ready for the next intense interval.

Now he hit skip again, and again, and again, until he gets another good, fast song that gets his heart rate back up. And it happens to be one of the best songs ever made, Free Bird! But this song starts out way too slow to get Jeff pumped up for a 30 second interval. He drags his finger across the scrubber to get to the good part, missing several times and having to redo it. By the time he gets it right, the interval is over and he's screwed up the whole workout.

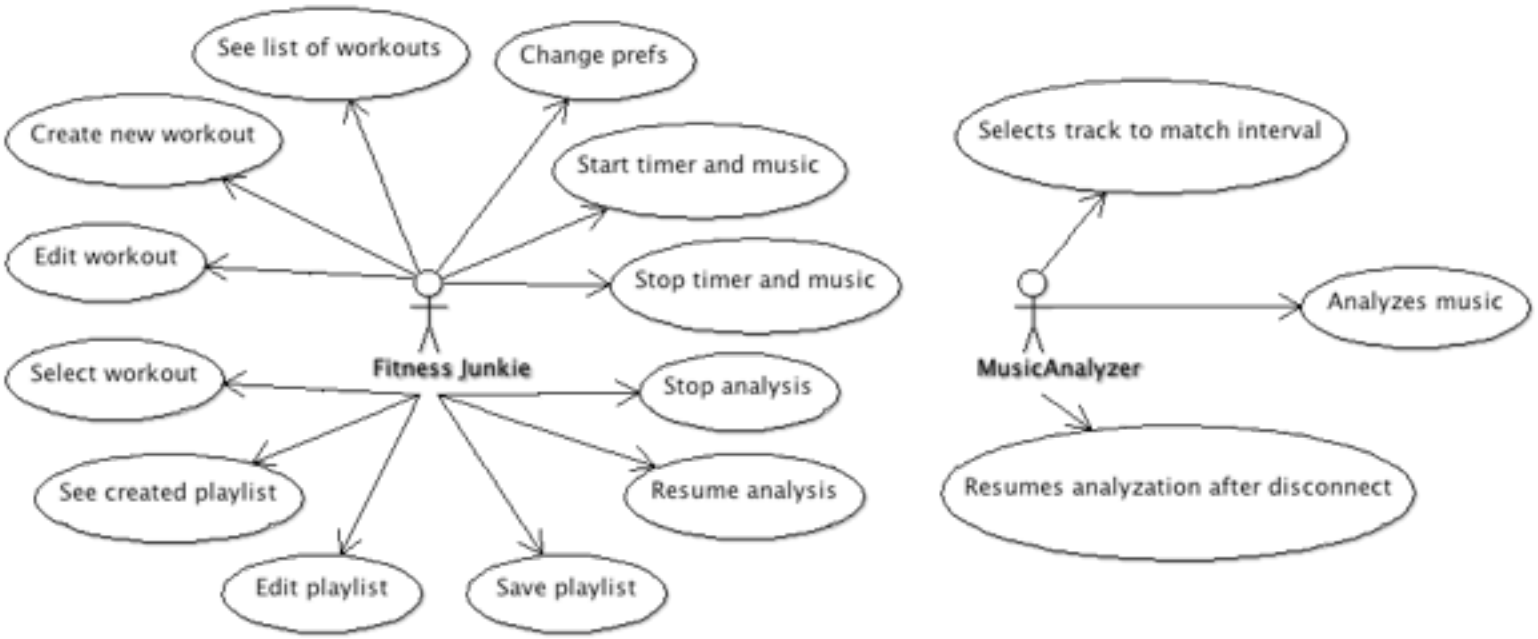
There's gotta be a better way, I know! He can make a playlist, but wait... no that still doesn't really work because the songs don't match the length of the intervals and he doesn't want to keep skipping and scrubbing in the middle of a workout. If only there was an app that created dynamic playlists using only the best parts of songs that match the tempo of your workout and help you hit your target heart rate every time.

Oh wait! There's that new app called HIITMix that does exactly that. He downloads it and lets it analyze his library while he goes to get that radio he kinda chucked out the window. Oops. When he gets back, the app has already created the perfect playlist for him! And it includes that awesome part in Bohemian Rhapsody on one of the intervals! And the best part, no commercials or Justin Beiber! Jeff McBroFist officially gives HIITMix two BroFists up.

Use Cases

1. Fitness junkie sees list of pre-made workout regimens.
2. Fitness junkie selects desired workout.
3. Fitness junkie sees playlist created for the selected workout.
4. Fitness junkie accepts playlist and sees timer, ready to start.
5. Fitness junkie starts timer and the app plays music.
6. Fitness junkie stops timer and music stops playing.
7. Fitness junkie resumes timer and music resumes.
8. Fitness junkie edits playlist until satisfied.
9. Fitness junkie saves playlist.
10. Fitness junkie creates new workout.
11. Fitness junkie edits existing workout.
12. Device's music library is updated, app begins analysis on new tracks upon next launch.
13. Network connection is lost during analysis, resumes automatically upon reconnection.
14. Fitness junkie cancels analysis.
15. Fitness junkie resumes analysis.
16. Fitness junkie changes preferences, analysis only allowed on Wifi.
17. Fitness junkie allows analysis on cellular network.
18. Fitness junkie selects saved playlist containing track no longer on device.

Use Case Diagram



Use Cases

Use Case #1: Fitness junkie starts timer and the app plays music.
This first and most important use case for this application is that it plays music along with the workout timer. This case assumes that a music file has been selected for each interval and assigned a time marker at which to begin playback. Once the user has started an interval timer, the application should begin playing the song assigned to the first interval. At the end of each interval, the song should fade out as the next song fades in. The volume should be controllable with the device's hardware volume switches, if they are available, and from within the application's user interface. Playback should behave as usual on the device, pausing when an event such as a phone call interrupts the application and continuing to play when the application becomes a background process. Two timers will be displayed on the screen, one for the total time and one for the current interval. The main timer will start at 0:00 and count up as the time progresses. The interval timer will start at the length of the current interval and count down to 0:00, at which time it will reset with the length of the next interval.

Use Case #2: App analyzes music files on the device.
The application should analyze the entire music library on the device the first time the app is launched. Upon each launch, the app should check to see if there are any music files that have no been analyzed. If so, it should begin analyzing the files using the device's network connection, if available. If the connection is not available or is too slow, the application should display a warning to the user that analysis could not be completed. If the user decides to ignore the warning and continue to use the application, it should attempt to match songs with intervals using only the files it has already analyzed. If there are insufficient analyzed songs, the app should simply use the device's shuffle feature when choosing tracks for intervals and set the playback marker to the beginning of the song.

Use Case #3: Fitness junkie creates new workout and edits existing workouts.
The user should be able to easily create a new interval set from scratch or possibly from a template. They should have complete control over the number, duration, and intensity of intervals. Once satisfied with their workout, it should be saved to the device's database of workouts. The user should also be able to edit or delete any existing workouts just as easily. Any playlists linked to these workouts will be unlinked, but not deleted so the user can modify them to fit back with their modified or new workouts.

Domain Objects

Workout - Container for a set of intervals that comprise a workout routine along with a link to any playlist made for that workout routine.

Track - Music file along with additional metadata.

Playlist - Collection of Tracks that is linked to a particular Workout.

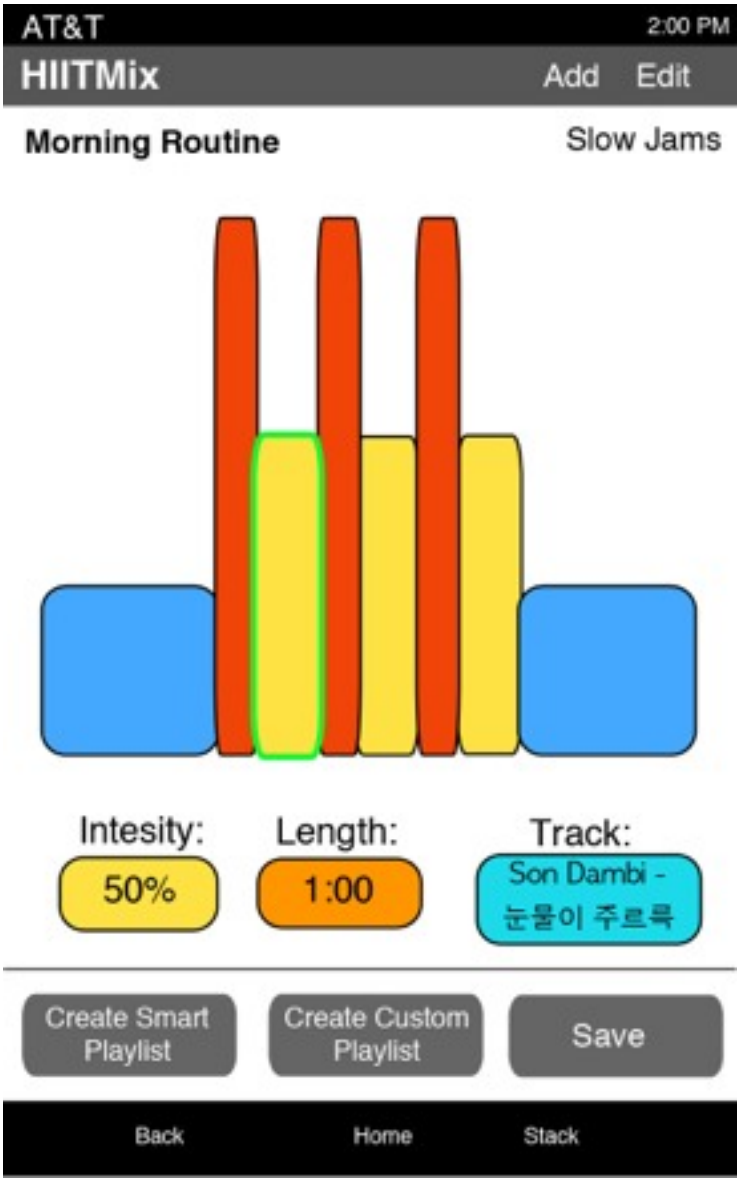
Interval - Unit of work in a Workout, comprised of intensity level and duration.

Timer - Simple timer indicating length of Workout or Interval remaining.

Screen Specifications

Workout List View - User is greeted with a list of available workouts when the application is launched. The available workouts are listed on the left with their linked playlists on the right if available. Contains options to add or edit workouts, both of which are handled in the Workout Edit View. Workouts are selectable, once one has been selected, clicking the start button will switch to the Timer View and begin the timer and music.

Workout Edit View - Shows current workout if user is editing a current workout or a default workout if the user is creating a new one. Displays individual intervals as vertical bars that represent the intensity (y-axis) and duration (x-axis) of the interval. User can set intensity, length, and track. Colors of intervals, lengths, and tracks indicate the intensity, duration, and tempo/energy respectively. Red and orange are more intense, longer, and faster tempo while blue and green are easier, shorter, and slower. Allows user to rename and save the workout.



Playlist View - Displays the songs in the current playlist. Contains options to add or edit songs in the playlist, as well as linking the playlist to a workout and saving the playlist for use later.

Timer View - Displays two timers, a large one for the interval and a smaller one for the overall workout time. Clicking the pause/resume button will pause or resume both timers as well as the music. Displays current song title and artist along with name of workout and playlist.