

# **Climb Application Proposal**

“Climb is a Fitness RPG. Battle your friends and earn powers as you stay healthy.”

Max Marze

## App Definition

Climb is a Fitness RPG. Battle your friends and earn powers as you stay healthy. Steps, exercise minutes, calories burned, sleep, and many other factors of your life can be used in battles to best your friends and level up.

## Features

Based on the app definition, the following features will be required for Climb to be considered feature complete,

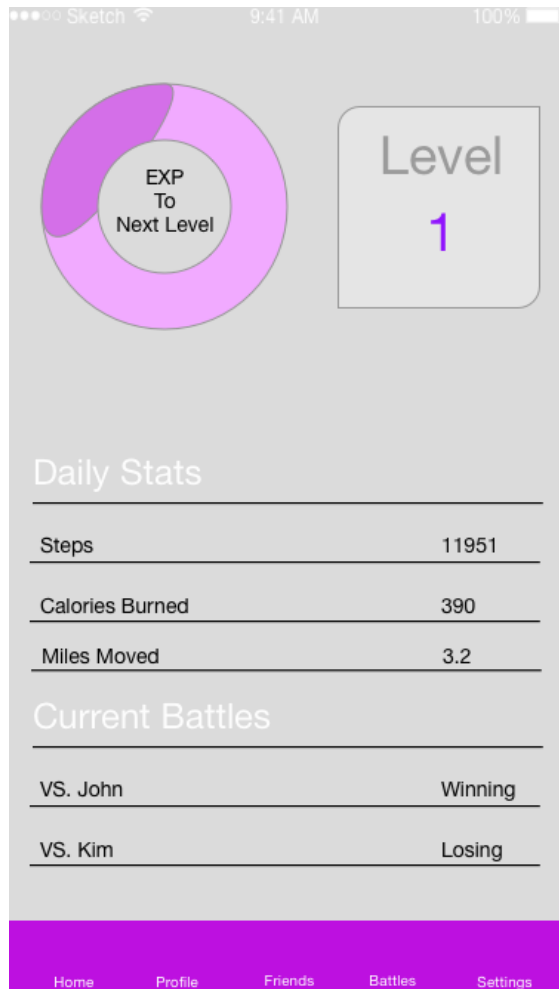
1. Battle with Friends through a Social Component
2. See where Friends are working out/exercising on a map
3. Integration with Apple Watch for collecting health information for battles. This is the essential feature of the app
4. Battles last 24 hours from when both users accept battle request
5. As you win battles you gain experience, which in turn allows you to level up
6. Leveling up gains power-up abilities, which gives players advantages against each other

## Navigation Flow

Climb will consist of 6 screens as follows,

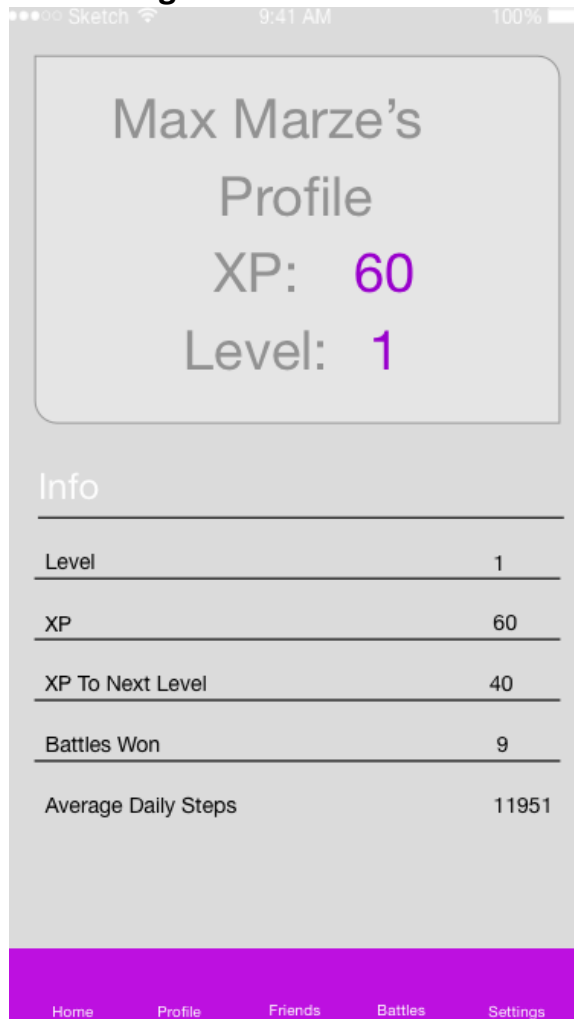
1. Main Screen: summary of The User's status, brief overview of battles currently being played
2. Profile Page: detailed view of The User's current status, history of battles and stats
3. Friends Page: information about The User's friends, The User challenges others to battles from this screen
4. Battle Page: all information about the currently selected battle, list of currently available Power-ups
5. Power Up Page: list of all the power ups The User has access to, list of power ups The User has equipped in a specific battle
6. Settings Page: allows The User to modify various settings, such as their display name and profile picture, as well as delete local cache if they want to force a resync of information

## Main Page



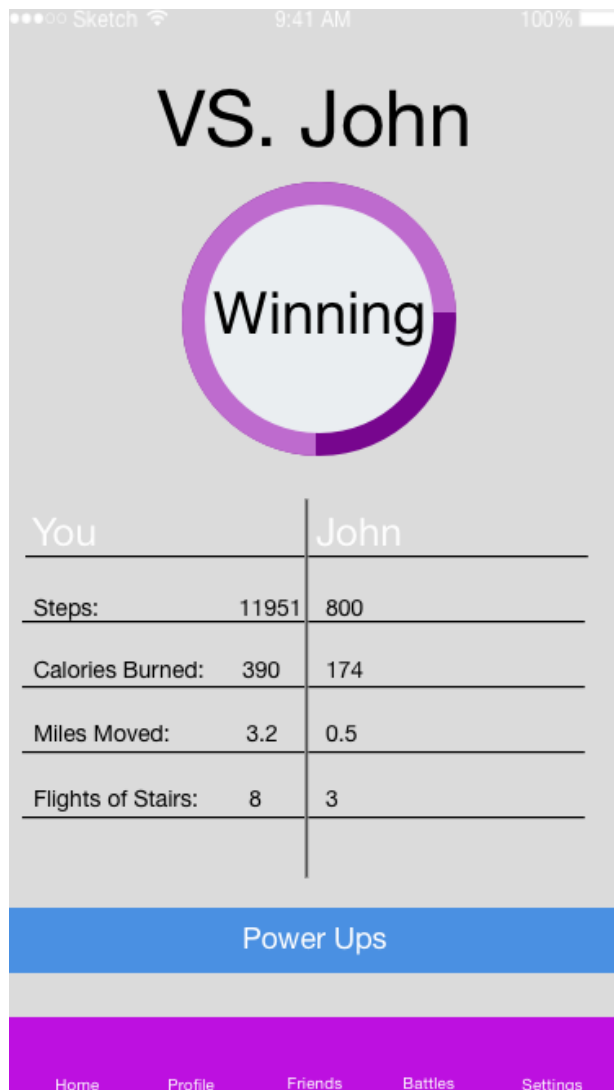
As shown here, the main page will focus on a high level overview of The User's life in Climb. Using data visualization, The User will see a Donut Chart depicting how much experience they have towards the next level. From the Main page, The User can navigate to the Profile, Friends, Battles, and Settings Pages.

## Profile Page




The Profile Page shows detailed information about The User. The Profile Page has access to The Home(Main), Friends, Battles, and Settings Pages.







## Battle Page



The Battle Page shows information about the Battle currently selected by The User. It shows the stats currently being considered in the battle. The Donut Chart at the top shows who is currently winning, proportional to the scores of both users. The Battle Page has access to the Power Ups, Home (main), Profile, Friends, and Settings Pages.

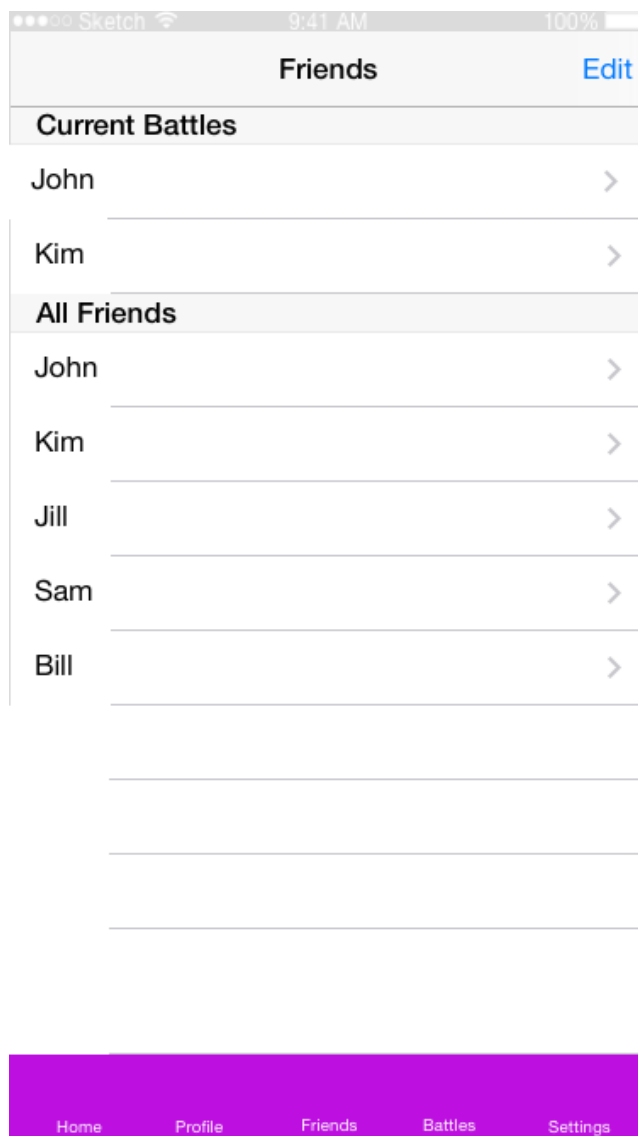
## Power Ups Page



< Back	Title	Edit
Equipped Power Ups		
	Power Up 1	>
	Power Up 2	>
	Power Up 3	>
Available Power Ups		
	Power Up 1	>
	Power Up 2	>
	Power Up 3	>

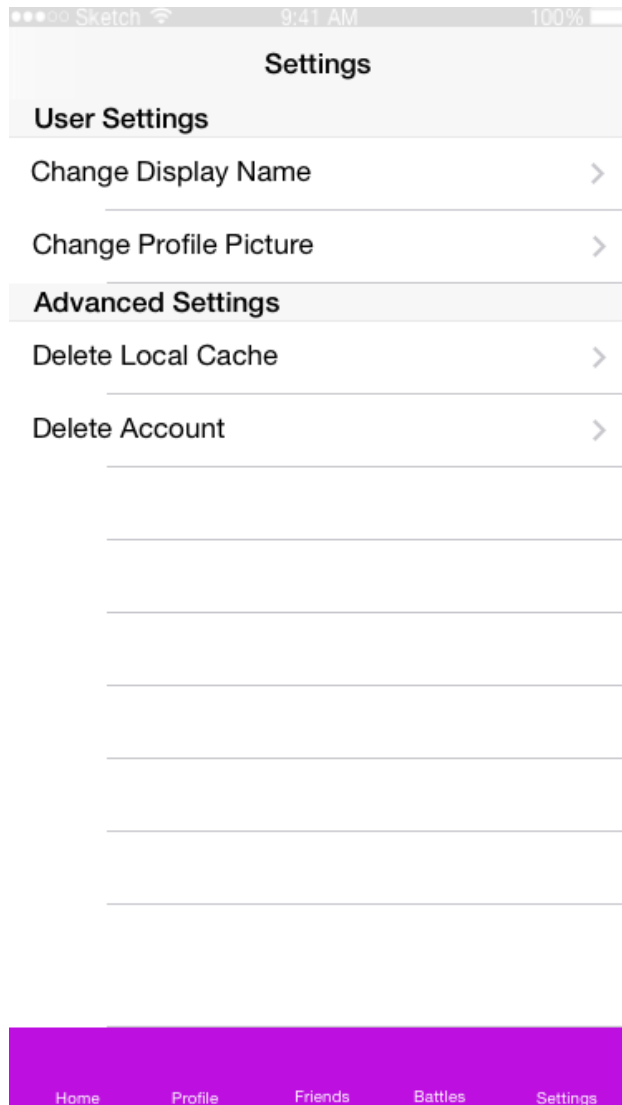
The Power Ups Page is a modally presented Table View, accessible from the Battle Page, that shows the currently equipped and available power ups for a specific battle. The User can equip and un-equip power ups by tapping the red or green selection box in each cell. This page's only navigation option is to go back to the battle page.

## Friends Page



The Friends Page contains all The Users Friends, sorted into Friends that are currently in battles with The User, and an All Friends section.

## Settings Page



The settings page contains all information related to changing user settings and preferences. There are also options if The User wishes to either delete the local data or delete their account. The Settings page can navigate to the Home (main), Profile, Friends, and Battles pages.



## Technologies

Based on the features of the app, Climb will utilize a combination of Apple Frameworks and API's, as well as a number of Third Party Libraries. To manage dependencies during Climb, Development Cocoapods will be used. This will allow for portability of development, as well as easy upgrades of the Third Party Libraries and Frameworks being used.

For access to Health Information, Climb will utilize Apple's HealthKit. An important factor to take into consideration when using HealthKit, is that it collects information from many sources other than an Apple Watch. To deal with this, the data will have to be filtered. This will ensure fairness of data that is used when battling another person.

The social aspect of the application will utilize Apple's Game Center. While Climb is not entirely a game, Game Center will provide the best possible social experience when finding friends to battle against. Game Center requires a paid developer account, which I currently have, so that will not be a problem.

Climb plans to utilize Charts and Graphs to better visualize trends in battles and fitness. To enable this functionality, a Third Party Library, 'ios-charts', will be used. The versatility of this library will allow Climb to take any information and visualize it with different types of charts.

To store information about Users and Battles, Parse will be used. It will allow Climb to maintain a synchronized database that will contain information about battles and users. Most information about battles will be stored in Parse so that both people will have access to synchronized data.

Mapbox will be used for displaying location information about friends and opponents. Mapbox is being used instead of MapKit because of the customization that is available for Mapbox. Climb can utilize custom maps, which will increase the overall user experience.

## Demonstration Plan

To demonstrate Climb, I will utilize real time data from my Apple Watch and then utilize random variations of my data with dummy users to demonstrate the social aspect. This will be the easiest way to demonstrate all aspects of the app in a relatively controlled testing environment.

## Development Plan

To make development as flexible as possible, Climb will utilize a Trello board where tasks and benchmarks will be organized. This organization will allow for prioritization of tasks. For example, the first goal is to have the xcode project set up with Cocoapods managing dependencies. Based on the features Climb requires, the first benchmark will be integration with HealthKit. Following that, completing Parse and Game Center integration will be the second milestone. UI elements will be implemented as needed when working on other steps.