Elliott Chartock (elboy@stanford.edu)
Veronica Cruz (vcruz27@stanford.edu)
Nolan Walsh (njwalsh@stanford.edu)
Monisha White (mewhite@stanford.edu)
CS194W
1/12/16

Project Proposal: The Workout Log

Description of the Project:
Our online social workout tracker will be a web-based application that works on both desktop and mobile devices. Users will login to our application, post workouts, and select the teams with which they want to share each workout. Entering a workout into our log will be easy enough that users can do so mid-workout, in order to encourage regular use and user retention. Users will also have access to all posts shared with the teams they are in, so they can follow their teammates workouts, and hold them accountable.

Key features of our application will include:

- The ability for users to login and post completed workouts that will be visible to the other members of their team(s)
- A quick and easy interface for users to create and add to posts mid-workout, as well as edit them afterward
- A social platform for users to engage in conversation with other users about workouts or other topics

Need for the Product:
There is a huge demand among athletes for an effective way to track their workouts. Athletes want an easy way to input lifts so they can look over their workout history and track their improvement.

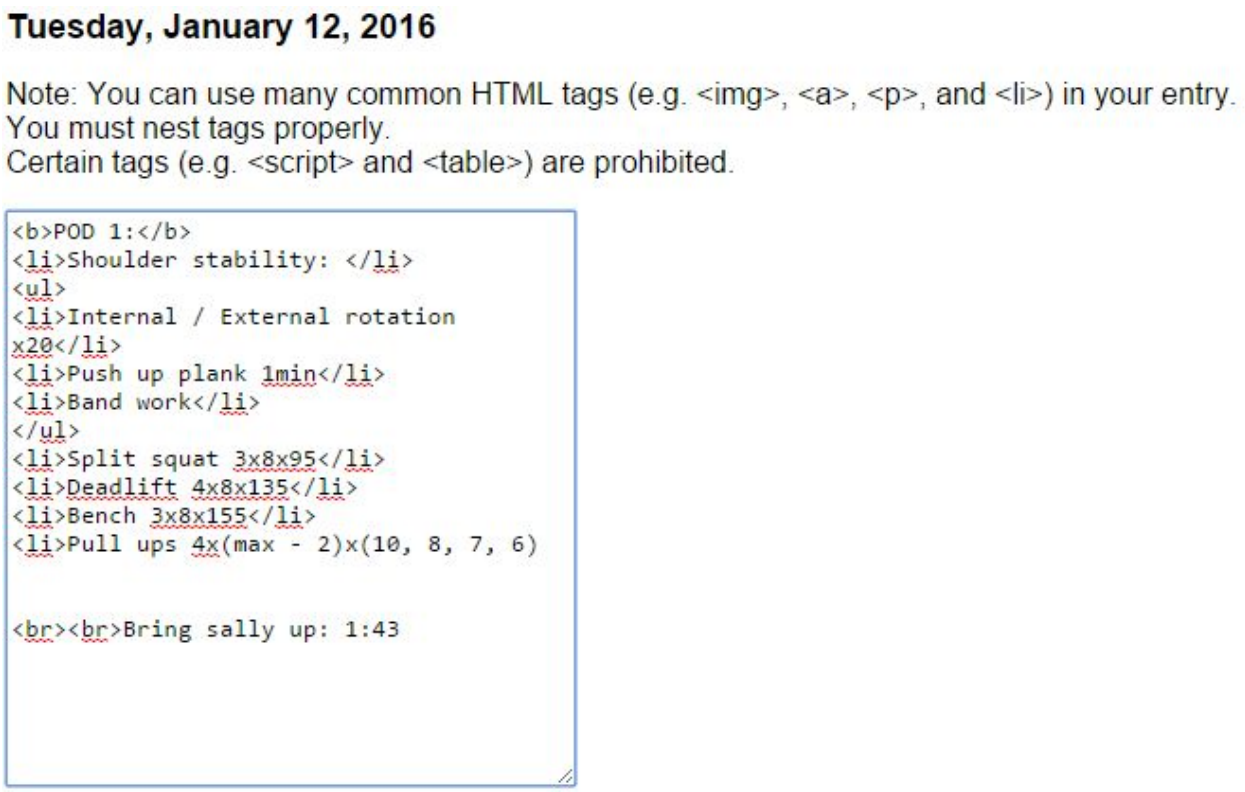
In addition, many teams (including the athletic teams of which we are members) are looking for ways to share their workout routines with each other, communicate about their workouts, and keep teammates accountable. Team leaders (captains and coaches) often want to ensure that all team members are completing required workouts and be aware of how much outside of practice work team members are putting in. Teammates can provide feedback or advice on workout routines if they have access to each other's workout posts. Users can also learn from their teammates' workouts, and get ideas on how to enhance their own workouts. We see the team aspect as a motivating factor for both working out and using the app.

Tracking reps and sets on paper does not easily allow progress to be seen over time. At the same time, using a digital tool in which a user has to type workouts or constantly edit text entries as he works out is cumbersome — many users would benefit from a digital workout tracker with an easy and quick way to select lifts and enter in the weight and reps performed. Additionally, our

app will contain accelerators that allow often-repeated or recently entered exercises to be more quickly selected by users.

Discussion of Competing Products:

There is already a website (workoutlogthing.com) that implements many of the features desired for this project. It allows users to create accounts, join teams, and post workouts that can be seen by other members of those teams. However, there are many issues with this site. Posting a workout consists of typing HTML-formatted text describing the workout, rather than the simple click-and-select UI that we hope to implement. This makes it tedious and time-consuming to write posts (especially on a mobile device) and requires users to have some knowledge of HTML. Workoutlogthing.com also does not allow users to select who can see each of their posts. Instead, all posts are visible to all members of every team of which a user is a member. This can lead to conflicts and sharing of sensitive data if people leave teams or join competing teams. Some users want certain posts to be visible to only one of their teams.



Posting a workout on workoutlogthing.com. Users must have knowledge of HTML tags in order to format the post. This also leads to a poor mobile experience since < and > characters are slow to type on mobile devices.

There are also many existing mobile apps that allow users to log workouts and keep track of their progress. However, the vast majority of these lack any social features. The logs exist on a single device and are not displayed to anyone other than the owner.

There are also some social network applications that are structured around working out. However, these applications stray from the simplicity of a classic workout log and have many cluttering features. For example, one of the most popular networks, called Fitocracy, is centered around the gamification of exercise. As users track their progress, they earn points and badges, and are encouraged to hire online personal trainers provided by the company. We believe that for a workout application to be effective, it must have a low barrier to entry. This means that the interface must be simple, and unnecessary extra features should be removed. The application must be simple enough that users can easily post their workouts in real-time, as they complete sets and reps of different exercises.

Potential Audience:

We are primarily targeting teams who want to track workouts together. The immediate audience for our project will be the Stanford women's and men's Ultimate Frisbee teams, who have a desire and need for an effective way to track workouts together and keep each other accountable. Once it has been tested and used by these teams, we would like to extend our audience to other athletic teams at Stanford, and eventually teams around the world.

Our product also has an audience beyond competitive teams — active people can use this product to create or support a social sphere around their workout routine. Having a group of friends who see each other's workouts and improvements can increase motivation and create a more positive exercise experience.

Our users need only basic familiarity with keyboards and web page navigation; no HTML or special text formatting knowledge is required.

Major Technologies Used:

- The Workout Log will be a web application built on Ruby on Rails. We primarily chose Ruby on Rails because of its notably well-written documentation. Furthermore, the simplicity of the convention-over-configuration design paradigm will be conducive to getting up and running quickly.
- For prototyping and user testing, there are a few tools that will be really helpful: POP (Prototype on Paper) is a great first-round digital prototype tool, because it allows users to feel like they are using a digital tool, but it only requires paper sketching as input; Balsamiq allows for prototypes that are a little rough but are still created quickly. Proto.io is extremely useful for figuring out the optimal flow of an app without investing heaps of time into coding different flow configurations; user testing with proto.io is effective because it looks similar to a finished product to the user. After flow and design have been determined, Photoshop is a great tool for creating pixel-perfect mockups, on which the specs of the code can be based.
- For coding the design layout, we will use Twitter Bootstrap and Sass, a powerful CSS preprocessing language. Bootstrap will be a useful tool towards creating a web app that is both mobile-first and responsive. Additionally, a Sass-powered version of Bootstrap is available for free on Github, which integrates with Rails easily since Rails comes

equipped with Sass by default. The official documentations for Bootstrap and Sass are helpful and thorough resources to use these tools effectively.

- While the Rails default database sqlite3 works well with single-user local applications, it does not work well for applications that require multiple users to access the same database. Since our application is a social platform that will indubitably connect many users to our centralized database, we will use a MySql database. DigitalOcean offers a thorough tutorial about how to set up a Ruby on Rails environment backed by a MySql database.

Resource Requirements:

Since our application is web-based, we will need a way to host it once we want to test it on a wider user base. Initially, we will be able to test the application locally, so we won't require web hosting until near the end of the quarter.

Potential Approaches:

We considered building our project as a mobile application instead of a web application. This approach has the advantage of guaranteeing an acceptable mobile experience. However, it would require building the application on both Android and iOS systems since different members of our teams use different mobile operating systems. We believe that by focusing on how our application looks and functions on mobile devices during development, we will ensure a good experience for both mobile and desktop users.

We also considered building an application that puts more emphasis on social interaction. This approach would center around users managing a list of friends with whom they want to share their workouts. It can be difficult to become motivated to workout alone, so putting a focus on social interaction does make sense. We decided that we want to put an emphasis on teams rather than friends, and design our application with that goal. Teammates want to motivate each other just as much as they want to hold each other accountable, so our application will allow users to do both.

Assessment of Risks:

This project requires knowledge of many different web-based technologies. While some members of our group have a bit of web development experience, we will all need to invest time in learning the technologies required. Learning new platforms and technologies has the potential to be a significant time sink. If this takes too long, we will certainly fall behind schedule.

Keeping our app secure could also be a challenge. We do not want to open up the app to a larger audience if it is not sufficiently secure. This project requires keeping potentially private user data, since users may post comments that they do not want publicly displayed. We need to minimize the risk of a breach of sensitive user data.

Next Steps:

A big next step for us will be familiarizing ourselves with the different technologies we have chosen. We have a bit of web development experience, but not enough to complete a project of this scale.

Our plan is to start by sitting down together and laying out a set of base features. After we figure that out, we will create mockups and prototypes of the flow of workout logging and other data entry, as well as logging in and joining/viewing teams. After we've found a desirable flow with low-cost methods will we begin to code what we've designed.