DreamTeam, Inc.

DreamTeam Athletics

Product Requirements Document

Prepared by:
Ben Hancock-Song
Henry Howe
Sebastian Olson
Conor Wellman

Release Date
November 30, 2023

Table of Contents

Document Versioning

Date	Owner	Comment
11/30/23	Team	Initial Draft

Project Description

It can be challenging to track a group's progress remotely. For example, coaches for college athletics teams lose the ability to train with athletes when they travel home for breaks. This can lead to people skipping workouts, overtraining, and greater risk of injury. Access to a community that shares your interests can increase motivation. Currently, there is a void of applications that encourage working out via an online community,

Our goal is to create a simple interface in which a user can input workouts they complete, track their progress over time, and share their profile with other users. The main screen of the GUI will be the user profile, containing basic statistics from past workouts such as workout time per day over the last week and totals for your most completed exercises . The main screen will also include an option to enter a new workout and will update accordingly after all information is collected.

A second goal for our project is to have a workout tracking function that runs during your workout instead of needing to manually input afterwards. We aim to, firstly, have a timer function that, upon stopping, adds the workout to the database. Secondly, we would like to add a GPS function that can track running or biking workouts. We would show users' locations on the map and track them through time and space. We would also like users to have the ability to attach comments and media to workouts that they can use to reflect later, including a pre-workout "readiness score" and images.

Another overarching goal for this project, most likely for a secondary iteration, is to implement an option to send your profile information to a designated account. This will allow users to compare progress and recreate some sense of community when working out alone. Additionally, it will enable sports teams and workout partners to track progress even from long distances. If time allows, a screen with more detailed information and exercise recommendations based on workout history and user goals could be implemented as well. To build the sense of connectedness within the app, another feature we hope to implement is workout and challenge assignments. This will allow coaches to assign workouts and friends to challenge each other to create a sense of community on the app.

Features

Feature Matrix

The feature matrix below describes all of the features outlined for this project. Each feature has an ID and Priority Ranking. Comments may be left to discuss feature progress. Striked through features have been dropped with reasoning discussed in comments.

Priority Codes:
H - High, a must have feature for the product to be viable and must be present for launch
M - Medium, a strongly desirable feature but product could launch without
L - Low, a feature that could be dropped if needed

ID	Priority	Feature Name	Owner	Comment	Case #
E.1	L	Login/Signup	Eng		
E.2	L	Add Friends	Eng		
E.3	L	Join teams	Eng		
E.4	L	Send Challenge	Eng		
E.5	H	Add/edit Workout	Eng		
UX.1	M	View Workout History	Design		
UX.2	H	GUI Support	Design		
UX.3	L	Workout Timer	Design		
E.6	L	Error Messages	Eng		
E.7	L	Workout Recommendation	Eng		
E.8	M	Share Workout	Eng		
E.9	H	Workout Stats	Eng		
E.10	L	Workout Calorie Tracker	Eng		
E.11	L	Filter Dictionary by Workout	Eng		

E.12	M	Error Handling	Eng		
UX.4	L	Workout Location Map	Design		
UX.5	M	Exercise Dictionary	Design		
UX.6	L	Friends Leaderboard	Design		

Feature Discussion

E.1 - Login/Signup

Users need to be able to log in to their account to access their data, or sign-up for an account to store data.

E.2 - Add Friends

Access list of other users and choose to create a connection. Adds connected users to a list of friends.

E.3 - Join Team

Access list of teams, which are groups of users. Users do not have to be friends to be a part of the same team. Users can request to join a team, and the creator/admin of a team can allow or dismiss their request.

E.4 - Send Challenge

Users can send pre-made workouts for their friends or teammates to complete. This will be especially useful for coaches to assign workouts to their teams.

E.5 - Add/Edit Workout

Users need to be able to log each workout they do. Workouts will be made up of a series of logged exercises. Users will have the option to add comments to their workouts. Workouts can be sorted into a type list. At the most basic level, users will manually input each element of each exercise and workout.

E.6 - Error Messages

If the application crashes, an error message appears in the GUI instead of the command line.

E.7 - Workout Recommendation

Based on previous workouts, the application recommends a next workout. For instance, if someone is skipping cardio, it may recommend a run.

E.8 - Share Workout

Send data from an individual workout with friends in a consistent format and receive/view workouts from other users in the same manner.

E.9 - Workout Stats

View basic statistics based on workouts entered over time, including cumulative workout duration for the last week and all time, workout streak, and exercises completed most often.

E.10 - Workout Calorie Tracker

Calorie tracker is an extension of the timer and workout type functions. It uses this data to calculate how many calories a person burns.

E.11 - Filter Dictionary by Workout

When accessing workout data in the dictionary, users can filter the dictionary by workout types. For instance, a user could just view their runs instead of their runs, bikerides, and lifts all together.

E.12 - Error Handling

If an error occurs, the application stops attempting to run. It either resolves the error or closes. Creates an error report that the technical team can recreate, evaluate, and fix.

UX.1 - View Workout History

View a list of users previously completed workouts in the order they were inputted. Information for each workout would include date, duration, and workout type. Clicking/hovering on a workout could show more detailed data.

UX.2 - GUI Support

Application runs on a GUI instead of in the command line. The GUI includes at minimum a main screen from which to navigate features.

UX.3 - Workout Timer

Rather than manually inputting workout duration, users can start/stop a timer within the app.

UX.4 - Workout Location Map

Users can show the location of their workout on a map.

UX.5 - Exercise Dictionary

The exercise dictionary will store a preloaded set of exercises for users to choose from. This will save time in logging workouts, since not everything will be manually entered. Exercises in the dictionary will be sorted by type.

UX.6 - Friends Leaderboard

Friends will be able to view each other's statistics over the time periods week and all time. Leaderboard statistics will include time worked out and workouts completed.

User Stories

The primary users of DreamTeam Athletics are athletes. Different types of athletes use DreamTeam Athletics. Users range from beginners looking for motivation and guidance, experienced athletes hoping to track progress and share with friends, and experts who diligently track each workout.

Rob: A 28 year old male just starting their fitness journey:

Rob is someone who has little to no experience working out in the past. Recently, their girlfriend broke up with them and Rob is now finally motivated to start exercising. Rob downloaded DreamTeam Athletics hoping for guidance on how to start, motivation to keep going, and a new community to be welcomed into.

Jimmy: 19 year old male on a competitive running team:

Jimmy has been exercising for most of his life and uses DreamTeam Athletics to track his exercises. He is part of a running team where his coach can both track the teams workouts and send workout plans to the entire team.

Boberta: 64 year old female who shares their swim workouts with their friends:

Boberta is an older lady who likes to swim to maintain good health. She has a close knit community of swimmers at her local pool that use DreamTeam Athletics to share workouts and promote friendly competition. After their weekly swim sessions they meet at their favorite vineyard and discuss the past week's happenings over wine and cheese.

Rory: 14 year old non-binary competitive crossfitter:

Rory has done crossfit their entire life. They use DreamTeam athletics to prove to the world the incredible athletic feats of strength that a 14 year old can perform. Otherwise, it would be hard to prove. DreamTeam's livetracking and team functions help share their lifts.

Use Cases

The following use cases describe the applications of DreamTeam Athletics' required features, error handling for potential failures, and possible extensions for future iterations.

Use Case 1

Name	Create account
ID	UC01
Description	User wants to create a new account
Actor	User
Organization Benefits	DreamTeam wants many users, each of whom will need to create a new account
Use Frequency	Very frequent
Triggers	User selects create account on the login screen
Preconditions	The application has loaded and displays the login screen
Main Success Scenario	● User selects create account ● User enters a name ● User enters a password ● System saves the new account
Extensions (Error Scenarios)	● User enters a name already associated with another account ○ System presents an error ○ System prompts user to enter a different username
Alternate Courses	● User selects an exit option on the create account prompt ● System returns to the login screen
Post Conditions	System logs into the new account and continues to the main screen

Use Case 2

Name	Log in
ID	UC02
Description	User wants to log in to their account
Actor	User
Organization Benefits	DreamTeam wants users to access their accounts many times
Use Frequency	Very frequent
Triggers	User selects log in on the login screen
Preconditions	The application has loaded and displays the login screen
Main Success Scenario	● User selects log in ● User enters a name ● User enters a password ● System logs in to the user's account
Extensions (Error Scenarios)	● User enters an unknown username ○ System displays an error message ○ System prompts the user to enter a different username ● User enters an incorrect password ○ System displays an error message ○ System prompts user to enter a different password
Alternate Courses	● User selects an exit option on the login prompt ● System returns to the login screen
Post Conditions	System logs into the entered account and continues to the main screen

Use Case 3

Name	Add Workout
ID	UC03
Description	User enters a new workout
Actor	User
Organization Benefits	DreamTeam will collect more data to better suggest future workouts
Use Frequency	Very frequent
Triggers	User clicks “add workout” button
Preconditions	User has an account and is on the home screen
Main Success Scenario	● User selects add workout ● User enters type of workout ● User enters duration of workout ● User enters distance ● Data is saved
Extensions (Error Scenarios)	● User enters incomplete information ● System presents an error
Alternate Courses	● User clicks out of pop-up ● System returns to home screen
Post Conditions	Exercise is saved and the user is returned to the home screen

Use Case 4

Name	Added friend
ID	UC04
Description	User added a friend, friend accepted

Actor	User/friend
Organization Benefits	Friends can invite more friends to use the service.
Use Frequency	Fairly often
Triggers	User searched for friend's name in list of users and clicked "send friend request"
Preconditions	Both friend and user owned accounts
Main Success Scenario	● User searches for friend ● User finds friend in user list ● User sends friend request ● Friend receives/accepts request ● Both users are added to eachothers' networks
Extensions (Error Scenarios)	● Friend does not show up in user list ● Friend does not receive request ● Accepting request fails to create friend connection
Alternate Courses	● Friend declines user request
Post Conditions	Users are saved as friends and can share their workouts.

Use Case 5

Name	Edit workout
ID	UC05
Description	User retroactively edits a previous workout
Actor	User
Organization Benefits	Users can add better data and change incorrect data.
Use Frequency	Lower
Triggers	User clicks on an existing workout in workout list.
Preconditions	User added a workout to their account.
Main Success Scenario	● User successfully adds workout ● User finds workout in workout list ● Clicking on workout brings up editable form with

	workout information - User changes entries in the form - User can save changes in the workout list
Extensions (Error Scenarios)	- Workout not found in workout list - Clicking on workout in list does not bring up form or brings up a blank form - User is unable to save changes or changes are saved as a new workout
Alternate Courses	- User cancels changes in the form
Post Conditions	Workout changes are saved and user returns to the homescreen

Use Case 6

Name	Check Leaderboard
ID	UC06
Description	User checks who has spent the most time working out in the past year
Actor	User
Organization Benefits	Creates community and friendly competition among users
Use Frequency	Occasional
Triggers	User clicks on leaderboard in friends tab
Preconditions	User has added friends to their network
Main Success Scenario	- From main page, user clicks on friends tab - User clicks on leaderboard tab - User selects year for time frame - User selects minutes worked out for statistic
Extensions (Error Scenarios)	- User has no friends added - Workout data does not update accurately, causing leaderboard data to be off
Alternate Courses	- User returns to main menu from friends page - User returns to friends page from leaderboard

Post Conditions	User is aware of leaderboard standings

Use Case 7

Name	Send challenge to friend
ID	UC07
Description	User sends a premade workout to an added friend
Actor	User/Coach
Organization Benefits	Builds community, adds functionality for athletic teams
Use Frequency	Often
Triggers	User clicks “Challenge” button next to friend’s username in friends tab
Preconditions	User has friend added
Main Success Scenario	● User goes to friends tab from main page ● User clicks “Challenge” button next to friend they would like to challenge ● User enters type of workout ● User enters duration ● User hits complete and sends workout
Extensions (Error Scenarios)	● User has no friends to challenge ● Challenge does not send
Alternate Courses	● Cancel in challenge menu to return to friends ● Return to main menu from friends tab
Post Conditions	Friend has new challenge sent to them