Lift On Up

Christopher McKenzie, Amisha Patel, Poojaben Patel, Ibrahim Alqarni, Guowei Li

9/15/21

**INTRODUCTION**

**This is a brief description of the project, at most 2 paragraphs. State what is the problem you are trying to solve and perhaps some user stories of how it will be used. Also describe success criteria.**

It will be a social fitness application with the purpose of motivating people to exercise and stay in shape. Users can add other users, create or follow workout routines, track their exercises, and track their progress. Additionally, users will be able to create routines by picking exercises from a list, categorized by muscles targeted. Users should also be able to view any of these exercises and the application will demonstrate how to perform the exercise. The mobile application will be accessible on iOS and Android devices.

**Example user stories:**

Jeff wants to get stronger but he has no significant experience with weight lifting. He downloads our app, Lift On Up, and he is able to find a list of workout plans. He uses the filters in the app to find the appropriate workout plan that caters to his work-life style.

Jenny is a fitness instructor. She is curious how well her workout routine compares to others and if there is room for improvement. She uses our app to create and publicize her routine. She receives ratings from 1-5 as well as feedback from other users who have tried her routine.

Justin used to workout with his friends but he moved to a new state for his new career. He prefers to workout in a group rather than alone. As a substitute, he downloads our app. With it, he is able to add his friends, see when they are online, and follow the same routines they used to do together. They are able to communicate with each other through private messages, stay in touch, and provide feedback to one another.

# ROLES AND RESPONSIBILITIES

**These vary for each type of product and, for small project teams, folks will serve multiple roles. This is a list of common roles:**

Project Lead - Chris

Project Manager - Ibrahim

Development Lead - Poojaben

Architect - Chris

Developers - Everyone

Infrastructure lead - Poojaben

Test Lead - Guowei

Testers - Everyone

Documentation - Everyone

Designer - Amisha

Risk/Requirements Management - Ibrahim

Customer Representative - Amisha

# METHOD

* Software:
  + **Describe the software languages, tools, frameworks and packages that you expect to use**
  + React native, iOS platform, Android platform, HTML, CSS, Javascript, MongoDB, SQLite, GitHub, Trello, Visual Studio, Android Studio, XCode, Firebase, Flutter
* Infrastructure:
  + **Describe the infrastructure that you expect to use for development, testing and production deployments**
  + Separate program for development, separate programs (3) for integration testing, unit testing, and stress testing, security testing
  + Main branch and separate branches for each sprint. Merge most recent branch to main branch after each successful sprint
* Development Process:
  + **Describe your high-level plan for development.Do you plan to do architecture, usability, design, security, privacy or code reviews?**
    - The goal is to follow a feature driven development plan and test features simultaneously with development. We shall perform code reviews after each sprint to do refactoring of the code, conduct daily standup meetings, and perform integration testing.
    - Create a new branch for each sprint. After all testing is successful for that branch, we will merge that branch into the main branch.
  + **Describe the testing that you expect to plan and execute, along with testing metrics that you plant to capture**
    - We will maintain all unit tests on a separate program from the app. This updated with new tests and ran every time when a new feature is added
    - Metrics will include size (Lines of Code), time (hours) taken to complete a feature, runtime complexity, date completed
  + **Describe any other metrics that you plan to keep track of over the duration of the project to measure progress.**
    - Create a product backlog, select the user stories(features) for each sprint, track the progress with burndown chart and task board managing applications like Trello, and regular commits on GitHub.
* Build Plan:
  + **Using GitHub for managing code and other project artifacts is mandatory. Describe how you intend to use the specific features that are provided with to support your project activities (e.g. branches, pull requests, actions, issues, …)**
    - Begin work by creating a Sprint 1 branch. When adding a new feature to the app, team members must create user tests within the user test program. Team members may only merge to that Sprint branch if all user tests run successfully. If there are any issues, they may comment out their new code if they want to merge it into the app in Sprint 1, otherwise, they may not merge. Once all features have been completed for Sprint 1 and all unit tests pass successfully, Sprint 1 will be merged into the main branch. Repeat the same process for each subsequent Sprint by creating a new branch for each new Sprint.

# COMMUNICATION PLAN

## Working team meetings

**These meetings form the core of the team’s collaboration and can be used for brainstorming, pair-programming, debugging, prototyping, internal reviews of each other’s work, etc.These are done without the manager’s (the instructor) involvement. Often it can be done by the team during the time of the scheduled class while other teams are having status meetings, but this may not be convenient if the team is a hybrid mix where some members are on campus while others are remote.**

**Whether you have your working meetings during class time or not, each team should also agree upon some recurring time(s) when to meet to work as a group in case it is needed.**

**State here what has been agreed upon for working team meetings**

Since two members are working remotely, during class time, working meetings will be held over Zoom in a separate room. If that is not possible, a Zoom meeting will be help immediately after class on Thursday.

## Status meetings

**Status meetings are meetings where the project manager and/or project leader reports on status to the instructor.These will occur weekly over the duration of the project during scheduled class time with the entire team present so they can all speak to any questions of status to the instructor.**

At the beginning, there will be stand up meetings every 3 days to discuss progress or any issues. After 2 weeks, we will hold these meetings on a weekly basis. Bi-weekly sprint review meetings to discuss sprint progress, code written, plan the next sprint, update product backlog. Bi-weekly sprint retrospective meetings to discuss what needs to be improved and how. All meetings will be held through Zoom.

## Issues meetings

**If a problem does arise, never surprise your manager. Schedule a meeting at his or her earliest convenience. This section describes how alerts will arise and the governance of when to trigger an alert – usually after a discussion at a working team meeting.**

Use Whatsapp to discuss any issues. If necessary, conduct a meeting through Zoom to look into the problem further. Any team members experiencing issues should notify the manager immediately, preferably not the day before a due date.

# TIMELINE AND MILESTONES

**Look forward to the next 10 to 13 weeks and plan out a set of target milestones that the team expects to meet over the duration of the project, culminating in the final presentation at the end of the semester.**

Counting every sprint as 2 weeks:

1. Sprint 1 - Infrastructure of the system
2. Sprint 2 Basic functionalities like login, registration, creating workout routines, tracking progress
3. Sprint 3 Advanced functionalities like adding friends, making profile public/private, rating routines, commenting, sharing, filtering system for searching workouts
4. Sprint 4 Stress testing
5. Sprint 5 Improve UI
6. Sprint 6 Cloud functionalities

# RISKS

**Describe any risks that you see for this project, and how they will be tracked and monitored.**

* The requirements might change.
* The technology used in this project may not provide the necessary functionality
* Majority of team has little to no working experience with most of technology used
* Making the project multi-platform may be difficult.
* The implementation of the project modules turned out to be more difficult than expected.

To avoid and monitor these risks, we will do the following:

* Apply agile methodologies to ensure the maximum involvement of all team members in every stage of the project.
* Taking the required training is a must to ensure that work is done on time.
* Organize the time and resources and prepare more meetings when it is needed.
* Document the status of work and progress every week. Progress will be documented in an excel sheet, which includes backlog, burndown chart, and sprints (size (Lines of Code), time (hours) taken to complete a feature, runtime complexity, date completed).

# ASSUMPTIONS

**State any assumptions you have about this project.**

* It will not be too difficult to make it cross-platform
* Security will not be a big concern because data only pertains to exercise
* Experimenting with new technologies will not be too time consuming; it should not interfere with meeting Sprint deadlines
* Ideally, each team member will successfully will implement two new features each sprint
* Each feature will include 3 unit tests
* We will have 100% coverage in our testing