**CS673 Software Engineering** 

**Team 3 - Health and Wellness Manager**

**Project Proposal and Planning**

| **Team Member** | **Role(s)** | **Signature** | **Date** |
| --- | --- | --- | --- |
| Chris Ceravolo | Team Lead | ***Chris Ceravolo*** | 9/23/2024 |
| Zihao Qian | Security | ***Zihao Qian*** | 9/23/2024 |
| Edward Lee | Configuration | ***Edward Lee*** | 9/23/2024 |
| Yu Luo | Design & Implementation | ***Yu Luo*** | 9/23/2024 |
| Amanda Yee | QA | ***Amanda Yee*** | 9/23/2024 |
| Kenny Light | Requirements | ***Kenny Light*** | 9/23/2024 |

**Revision history**

| **Version** | **Author** | **Date** | **Change** |
| --- | --- | --- | --- |
| 0 | Team 3 | 9/11/2024 | All sections |
| 1 | Amanda Yee | 9/21/2024 | Updated ‘Functional Requirements’ based on Samantha’s feedback |
| 1 | Edward Lee | 9/23/2024 | Update to officially decide on Heroku |
| 2 | Edward Lee | 10/06/2024 | Update Configuration Management |
| 2 | Chris Ceravolo | 10/07/2024 | - Update requirements  - Update management plan |
|  |  |  |  |

[Overview](#_heading=h.gjdgxs)

[Related Work](#_heading=h.30j0zll)

[Proposed High level Requirements](#_heading=h.1fob9te)

[Management Plan](#_heading=h.3znysh7)

[Objectives and Priorities](#_heading=h.2et92p0)

[Risk Management (need to be updated constantly)](#_heading=h.tyjcwt)

[Timeline (need to be updated at the end of each iteration)](#_heading=h.3dy6vkm)

[Configuration Management Plan](#_heading=h.1t3h5sf)

[Tools](#_heading=h.4d34og8)

[Deployment Plan if applicable](#_heading=h.2s8eyo1)

[Quality Assurance Plan](#_heading=h.17dp8vu)

[Metrics](#_heading=h.3rdcrjn)

[Code Review Process](#_heading=h.26in1rg)

[Testing](#_heading=h.lnxbz9)

[Defect Management](#_heading=h.35nkun2)

[References](#_heading=h.1ksv4uv)

[Glossary](#_heading=h.44sinio)

# Overview

*(Please give an overview of your project. It should include the motivation, the purpose and the potential users of the proposed software system, the basic functionality of the proposed software system and the possible technology stack to be used.)*

**Title:**

* Health and Wellness Manager

**Motivation:**

* There is a growing awareness of the importance of taking care of your health and wellness. It can be easy to lose track of our own diet, exercise, and wellness routines when life gets busy. This project seeks to provide a straightforward and user-friendly system that helps users stay accountable, motivated and on track.

**Potential users:**

* Anybody who wants to better track their personal health and wellness data
* Potentially additional user-types such as personal trainers and nutritionists

**Description:**

* This project is a web application designed to help users track their health and wellness, such as steps taken, water intake, sleep, exercise routines, and diet. Users can set daily / weekly goals in each of these areas and monitor their progress toward achieving their goals.

**Tech Stack:**

* **Front end:** React
* **Back end:** Node.JS, express.js, mongo database cloud connection
* **Web Deployment:** Heroku
* **Management software:** Jira
* **Testing:** Jest

# **Related Work**

*(Please describe any similar software systems that you have found through the online research, and the differences between your software and those software systems.)*

* **MyFitnessPal**
  + There are many existing health and wellness apps currently in the market. A lot of these apps, like MyFitnessPal, have a lot of functionality which can sometimes feel overwhelming due to excessive features. This project will focus on simplicity and ease of use while still providing essential functionality, and we can use existing apps to get ideas of features to incorporate.
* **Duolingo**
  + We will take inspiration from Duolingo engagement features—such as providing healthy competition and sending reminders and messages of encouragement to the user
* **Health (Apple)**
  + We can learn from Apple Health for its simple, clean and intuitive interface also uses simple, easy-to-read graphs to visualize data, which helps user navigate and understand their health information.

# Proposed High level Requirements

# [**Jira Board Link**](https://seprojects-cs673olf24team3.atlassian.net/jira/software/projects/HW/boards/1/timeline?shared=&atlOrigin=eyJpIjoiNWZjMTE5ZDBhMTI3NGZiYzlmMjRmNzk1ZmZmYTMzMjEiLCJwIjoiaiJ9)

* 1. **Functional Requirements**  
     (For each functional requirement, please give a feature title and a brief description using the following format: As (a role), I want to (action), so that (value).)
     1. **Essential Features** (the core features that you definitely need to finish):

(For each essential features, please give a rough estimation in terms of person hours or an range of person hours)

* Login
  + Parent feature: Manage Profile
  + Description: As a user, I want to log in so that my data is visible only to me.
  + Hours: 12
* Create Profile
  + Parent feature: Manage Profile
  + Description: As a user, I want to create a profile so that my data persists between logins and I can keep track of my progress.
  + Hours: 8
* Edit Profile
  + Parent feature: Manage Profile
  + Description: As a user, I want to edit my profile to update any details (e.g. name, gender) so that it reflects my current information.
  + Hours: 8
* Goal Form
  + Parent feature: Manage Goals
  + Description: As a user, I want to set goals so that I can have something to work toward.
  + Hours: 8
* Edit Goals
  + Parent feature:Manage Goals
  + Description: As a user, I want to be able to edit my goals to allow myself the flexibility to adjust the rigor of my wellness routines.
  + Hours: 8
* Submit Daily Summary Health Data Form
  + Parent feature: Manage Health Data
  + Description: As a user, I want to submit a daily summary of my health data (such as exercise and sleep) so that I can track my daily progress toward my wellness goals.
  + Hours: 8
* Edit Health Data
  + Parent feature: Manage Health Data
  + Description: As a user I want to be able to edit the health data I submitted so that I can fix any mistakes and more accurately capture my wellness routine.
  + Hours: 8
* Time Series Dataviz
  + Parent feature: Progress Dashboard
  + Description: As a user I want to see the trends in my health metrics over time so that I can repeat behaviors causing good trends and avoid behaviors causing bad trends.
  + Hours: 12
    1. **Desirable Features** (the nice features that you really want to have too):
       - Search and Add Friends:
         1. Parent feature: Join Community
         2. Description: As a user, I want to add friends so that I can share my progress towards my goals and encourage others to progress toward their goals.
         3. Hours: 2
       - Notifications Receiver Panel
         1. Parent feature: Receive Encouragement
         2. Description: As a user, I want to receive notifications of encouragement to help me remember to work towards my goals when life gets busy.
         3. Hours: 2
       - Streak Tracker Notifications
         1. Parent feature: Receive Encouragement
         2. Description: As a user, I want to be warned when I am about to lose a streak so I can stay engaged with my wellness routine.
         3. Hours: 2
       - Edit Profile Visibility
         1. Parent feature: Customize Settings
         2. Description: As a user, I want to be able to control what information other users on the app can see about me so that I can keep my personal health data secure.
         3. Hours: 2
       - Detailed Goal Statistics
  + Parent feature: Progress Dashboard
  + Description: As a user I would like to view my progress towards specialized categories for nutrition and workout goals so that I can assess the progress of my wellness routine at a finer resolution.
    - Example: % Distribution of workouts across strength, flexibility, and endurance
    - Example: % Focus on areas of body
  + Hours: 12
    - * Task Reminders
  + Parent feature: Receive Encouragement
  + Description: As a user, I want to be reminded when I have a task due to complete a goal so that I can avoid losing progress on my wellness routine.
  + Hours: 2
    1. **Optional Features** (additional cool features that you want to have if there is time):
       - Create Custom Workouts
  + Parent feature: Fitness Form
  + Description: As a user, I would like to set goals around my own workouts that are not available for me to choose in the app so that I can fully integrate the Health and Wellness Manager with my routine.
    - There will be specialized data associated with each Workout object, perhaps with scores ranking focus on parts of the body or other special data. So user may need to input some of this custom info in order to create a custom workout.
  + Hours: 2
* Dashboard Elements Adjustable by Timeframe:
  + Parent feature: Progress Dashboard
  + Description: As a user, I would like to adjust the scale of the progress dashboard (from daily, to weekly, to monthly, etc.) so that I can gain a more comprehensive view of how my wellness routine has progressed.
  + Hours: 2
* Nutrition Form
  + Parent feature: Manage Health Data
  + Description: As a user, I want to document what I eat and drink quickly, without entering too much data, so that I can easily track my progress toward my dietary goals.
  + Hours: 8
* Fitness Form
  + Parent feature: Manage Health Data
  + Description: As a user, I want to document my exercises quickly, without entering too much data, so that I can easily track my progress toward my fitness goals.
  + Hours: 8
* Competitive Leaderboard
  + Parent feature: Join Community
  + Description: As a user I would like to engage in healthy competition with other users of the Health and Wellness Manager so that I may be inspired to workout more and eat healthy.
  + Hours: 12
* Suggest Daily Calories and Workouts
  + Parent feature: Receive Suggestions
  + Description: As a user, I want to receive goal suggestions from the app so that I can explore unexpected possibilities for my wellness routine.
  + Hours: 5
* Specialized Preferences on User Profile:
  + Parent feature: Create Profile
  + Description: As a user I want to be able to set specialized preferences in my user profile so that the app can create more customized suggestions for my wellness routine.
    - Example: Preference for cardio workouts
  + Hours: 2
* Specialized Goal Categories:
  + Parent feature: Goal Form
  + Description: As a user, I want to be able to set specialized goal categories to customize my experience and align my
    - Example: % fresh fruits and vegetables
    - Example: Targeting exercises by strength, endurance, or flexibility
    - Example: Targeting exercises by part of the body
    - Creating specialized categories will require coordination with the overall UML diagrams, and other aspects of the project such as data visualization
  + Hours: 2
* Add Friends to Goals:
  + Parent feature: Goal Form
  + Description: As a user I would like to collaborate with goals on my friends to increase the fun factor of my wellness routine and help to hold myself accountable.
  + Hours: 2
* Smart Nutrition Data Entry
  + Parent feature: Nutrition Form
  + Description: As a user, I want the app to find the nutritional value of things that I eat so that I don’t have to manually enter calories for every single meal.
    - Connection to USDA data
    - Enter barcode number and get data from https://world.openfoodfacts.org/data
  + Hours: 5
* Goals Calendar
  + Parent feature: Progress Dashboard
  + Description: As a user, I want to check a calendar so that I can see what goals I have achieved in the past and plan toward the goals I have scheduled in the future.
  + Hours: 12
  1. **Nonfunctional Requirements**
     1. **Security requirements**
        + Encrypt user email and passwords in database (Hash Functions)
        + Use strong password requirements
        + Use HTTPS
        + “Sanitize” form input data to prevent cross site scripting (XSS) attacks
        + Manage user access if implementing multiple roles
        + Minimize data collection and avoid collecting unnecessary data

# **Management Plan**

## Objectives and Priorities

*(Please describe your project objectives with highest priority first. Project Goals can include but not limited to complete all proposed (essential) features, deploy the software successfully, the software has no known bugs, maintain high quality, etc )*

* We will aim to complete all proposed essential features.
* We will aim to complete some desirable features as time allows.
* Optional features will likely not be completed, but will provide useful activities in the event that we have more time than we anticipate.
* We are aiming to deploy this software as a web application.
* While we aim to produce a polished user interface, we will only design a responsive interface if there is time. More than likely, this will be a statically-sized webpage designed for desktop.
* We will aim to have no known bugs in the final product.

## Risk Management (need to be updated constantly)

*(Please write a summary paragraph about the main risks your group identified and how you plan to manage these risks. Then use the separate google sheet for detailed risk management. The template is provided in the same folder with this file. Please provide the link to the sheet.)*

As a team, we have identified several key risks for our project. One major risk is if one or more team members drops out from the class as that could affect our ability to meet the project goals we have set. Balancing schedules is also a concern, as each team member’s home and work life could interfere. We plan to keep constant communication via Discord if any scheduling conflicts arise and reschedule where necessary.

Given the six week project timeline, there is also a major risk that we will not achieve our goals due to scope creep, unclear requirements, and improper planning. We plan to mitigate these risks by having a clear project plan available to all team members via Jira.

In addition, there is technology competence risk with team members having varying levels of experience with React, Git and unit testing which could slow down development. We plan to individually upskill on these and also support each other through this learning process.

**Risk Management Sheet Link:** https://docs.google.com/spreadsheets/d/1B9yetc2xfoqSk5BoeNpZyjAfAHsmus2A/edit?usp=drive\_link&ouid=106673150530746711200&rtpof=true&sd=true

**Updates:**

* Closed “Management: Improper Planning”
  + We have completed the majority of our application. In the next week there remains only fine tuning of details, or the addition of desirable/optional features. We effectively adjusted scope as necessary throughout the design and implementation process. Automated deployment helped us to see what was possible along the way.
* Closed “Technology Competence: Not familiar with React”
  + All of us have committed code and used React successfully
* Closed “Technology Competence: Not familiar with unit testing”
  + All of us have written tests. Our application must pass tests in order to deploy.
* Closed “Technology Competence: Not familiar with Git”
  + We are all using Git as we contribute to our repository on GitHub
* Closed “Integration & Deployment (I&D): Not enough time for I&D”
  + We set up an automated pipeline and have been testing deployment throughout the development process. Our final application will deploy.
* Closed “I&D: Hosting Service”
  + We are using Heroku successfully

## Timeline (this section should be filled in iteration 0 and updated at the end of each later iteration) [Jira Board Link](https://seprojects-cs673olf24team3.atlassian.net/jira/software/projects/HW/boards/1/timeline?shared=&atlOrigin=eyJpIjoiNWZjMTE5ZDBhMTI3NGZiYzlmMjRmNzk1ZmZmYTMzMjEiLCJwIjoiaiJ9)

Requirements in this timeline are summarized at the highest parent level (e.g. Manage Profile includes Log In, Create Profile, and Edit Profile). Please refer to the “Proposed High Level Requirements” section or Jira for a more detailed breakdown.

| **Iteration** | **Functional Requirements(Essential/Disable/Option)** | **Tasks (Cross requirements tasks)** | **Estimated/real person hours** |
| --- | --- | --- | --- |
| 0 | - Planning & Analysis | Establish project idea and requirements  Define tech stack  Assign roles  Set up Jira  Set up Figma board | 40 hours |
| 1 | - Manage Profile  - Manage Goals  - Manage Health Data | Create UML diagram  Database schema  Set up Cloud hosting  Test deployment of a simple page  Demonstrate connection to database  Establish testing  Set up Docker  CI/CD | 135 hours |
| 2 | - Login/Logout  - Progress Dashboard (homepage)  - Refinement of Manage Profile, Manage Goals, and Manage Health Data | - Created React application  - Set up MongoDB database and defined initial schemas  - Created pages for Add User, View Users, Add Goals that connect to external DB  - Created screens for Login and Add Daily Health Data Entry  - Set up Docker  - Set up Jest testing framework and created initial unit tests | 140 hours |
| 3 | - Desirable/optional features as people show interest, and as time allows | - Refinement of manage goals  - Increased password security  - Generate dummy goals and health data for visualizing charts  - hook up charts to data  - write additional client and server tests to cover code entirely  - fix any bugs  - prepare final presentation | 60 hours |

# **Configuration Management Plan**

## Tools

*(In this project, we will use Git and Github as the version control tools. Please also specify any other tools to be used, e.g. IDE tools, CI/CD tools, container tools, SAST or DAST tools, and any other DevOps tools)*

In this project, we will use GitHub for version control. When new changes are pushed to the “main” branch, we will trigger GitHub actions to deploy the changes in the application. The GitHub plugin can also be used for user convenience in the IDE of their choice.

We will be using Figma to design the UI for our Health and Wellness Manager web app. By using Figma, we can visualize the overall structure and user interface of the web app, ensuring that the design is user-friendly and consistent.

* 1. **Code Commit Guideline and Git Branching Strategy***(Please briefly describe criteria for the code commitment and the branching strategy used, e.g. what are the branches to be used, how the pull request will be used etc. Here is an article to give you some basic knowledge about different git branching strategies:* [*https://www.flagship.io/git-branching-strategies/*](https://www.flagship.io/git-branching-strategies/)*)*

A “dev” branch will be used for all new features. Each individual team member will create a branch off of this branch (ex. “elee-ticket-id”) and merge their new change in. Tested features ready for use will be merged to the “main” branch by creating a release branch (ex. release1). A check is placed on pull requests to main from dev, where the application must successfully deploy to Heroku.

Pull requests will contain the Jira ticket ID and a summary of the feature. Hotfix pull requests to “main” will have the “Hotfix” label applied to them, with the defect and summary of the fix being in the title.

## Deployment Plan if applicable

*(If you plan to deploy your application (e.g. your web application), briefly describe how you plan to deploy your application).*

Pushes to the “main” and “dev” branches will trigger deployment through GitHub actions to Heroku.

# Quality Assurance Plan

## Metrics

*(Describe the metrics to be used in the project to measure the quality of your software. Each metric should be measurable and quantifiable. Examples of metrics include product complexity (LOC, # of files, # of classes, # methods, cyclomatic complexity, etc.), defect rate (# of defect per KLOC), # of test cases, test case pass rate, cost (# of person hours used), # of user stories completed, etc.* ***The result of these metrics should be reported in the progress report/ iteration summary sheet****)*

| **Metric Name** | **Description** |
| --- | --- |
| Lines of Code | Measures the total number of lines of code written for out project |
| File Count | Measures the total number of files created |
| Class Count | Measures the total number of classes created and used |
| Method Count | Measures the total number of methods created and used |
| Test Case Count | Measures the total number of test cases created |
| Jira Ticket Completion Count | Measures the total number of Jira tickets completed by the team |
| Cost | Measures the collective number of hours spent by the team on developing the web application |

* 1. Coding Standard

*(Describe any coding standard to be used)*

* + 1. camelCase for variables, functions, and object properties
    2. PascalCase for classes
    3. UPPER\_CASE for constants
    4. lowercase-kabob for file names
    5. Write clear and concise comments

## Code Review Process

*(Everyone should review all documents to be submitted. Here you will mainly describe how the code review will be done. Who will review the code, e.g. design or implementation leader will review all code or team members review each other’s code. Do you use pull requests for the code review? Is there a checklist to help review? What feedback should the reviewer provide?)*

Pull requests (PR) will be required for a code review and each PR will require **at least one** approval by another team member. When a PR is ready to be reviewed, post in the #team3 Discord channel and someone in the team should respond that they will review it.

The reviewer should ensure that the pull request fulfills the requirements of the related Jira ticket. The reviewer should also provide feedback if they foresee the code changes having negative downstream impacts.

## Testing

*(Both manual testing and automated testing should be considered. Both unit testing and integration testing should be considered. Briefly describe the testing tools/framework to be used, the personnel involved (e.g. the QA leader will focus on the integration testing and each developer will unit test their own code), when and what types of testing will be performed, the testing objectives, etc)*

Each developer will be responsible for writing unit tests for classes/methods that they develop. The objective here is to catch errors early in the development process and ensure that each piece of the application functions as expected in isolation. We plan to use Jest.

Integration testing will be considered with the objective to oversee the interaction between components and ensure the application as a whole is running as expected. For example, we will need to verify that the data flowing from the back end to the front end is working properly. The QA leader will be responsible for this and this will be done manually before the release of each project iteration.

## Defect Management

*(Describe the tool to be used to manage the defect (e.g github issues). The types of defects to look at. The actions or personnel for defect management.)*

To manage any defects in our application, we will use Jira to track them by creating a new ticket for the given defect. The main type of defects we will be looking out for are functional defects where a feature/functionality is not working as expected. UI/UX defects will also be considered for any issues related to the user interface or user experience.

For defect management, when a team member identifies a defect they should immediately record the defect via a Jira ticket that contains a description of the problem and a proposed fix. The team leader will prioritize defect handling, based on severity and downstream impact, and assign a team member to handle the ticket. The developer will work on the ticket, make the fix and commit the necessary changes to the codebase. The PR reviewer will verify the defect is fixed before approving the changes.

# **References**

*(For more details, please refer to the encounter example in the book or the software version of the documents posted on blackboard.)*

* Coding Standards: <https://www.syncfusion.com/blogs/post/top-javascript-naming-convention>

# **Glossary**

*(Any acronym used in the document should be explained here)*

N/A