Spotter

McMaster University

System Requirements
SE 4G06 & TRON 4TB6

Group 12

Donisius Wigie - 400143340
Zuhair Makda - 400065864
Artemiy Kokhanov - 400074430
Ridhwan Chowdhury - 400075426
Juwon Adeola - 400075635
Winnie Liang - 400074498

Table of Contents

Revisions

Date	Revision Number	Authors	Comments
October 15th 2021	Revision 0	Donisius Wigie	Added Monitored and Controlled variables
October 15th 2021	Revision 0	Donisius Wigie Winnie Liang	Added User Diagram and Data flow Diagram
October 17th 2021	Revision 0	Artemiy Kokhanov Ridhwan Chowdhury Winnie Liang	Added Functional and NonFunctional Requirements. Also added variables and diagrams
October 18th 2021	Revision 0	Juwon Adeola Zuhair Makda Ridhwan Chowdhury	Added Context Diagram and Normal Operation Description
October 18th 2021	Revision 0	Artemiy Kokhanov Ridhwan Chowdhury Winnie Liang Donisius Wigie Juwon Adeola Zuhair Makda	Final touch-ups and checks for submission

The Purpose of the Project

The purpose of this project is to build an application that allows users to train themselves correctly without having to depend on gyms and/or personal trainers. For many users, it is difficult to gage yourself whether your form is correct and unless told otherwise, the user may continue their workout improperly. S*potter* will assist users who are looking to begin a workout regimen by recommending useful exercises and guiding them through the workout with proper form. One of the most common problems with beginners who start to exercise is the high possibility of injury due to the lack of proper form. Spotter, however, provides a self-sufficient and cost effective solution that permits users to improve upon their exercises at their own pace without any additional time and money being spent.

Scope

This system is meant to be implemented locally by users to work out at the comfort of their own space and time. The user will be able to start up this application with a browser of their choice in tandem with using a webcam, starting their respective workout and performing their exercises for the session. On the application, they will choose the exercise that they're looking to perform, and will be able to track, correct, and improve on their form and rep count per exercise in real time. Once they've completed the exercise the application will display the recorded video of the user performing the exercise with the existing overlays illustrating how well the exercise has been performed and stores the number of correct reps for the user to track their progress. Once they've completed all the exercises and finished their session, they can close the application.

Normal Operation

The user looking to begin their personal gym session will start up the Spotter application locally on their computer/laptop of choice. The application will then display an interface in which the user will select the exercise they're planning to perform. Once that is chosen, they can select the "begin tracking" option from the interface and begin tracking, and performing the exercise in front of their webcam. The application starts to capture video of the user with the overlay constraints for the exercise, and the user performs the exercise. As the user performs a correct repetition of the exercise, a count of each repetition is stored and displayed as an overlay in the video. Once the user has completed 1 set of repetitions, they can end the video capture and begin their rest time. The user then repeats the same sequence until they've fully completed all the sets for the respective exercise. When the user wants to perform another exercise, they select the exercise within the initial Spotter interface and repeat the steps mentioned above.

Use Case Scenarios (U.x)

U.1: Application is started/initialized

The application, *Spotter*, starts when the user navigates to the designated webpage. Upon initiating, the user is prompted with a few options where the application awaits action from the user to begin normal operation.

U.2: Workout is started/requested

Once past the initial phase, the user can see themselves through a webcam or camera on the application. Based on the user's workout regimen, the application is to follow and track the user's movements and position of key joints/markers throughout the exercises warning the user if their form/posture has strayed away from the appropriate motion. The application will also keep track of the user's progress by keeping a count of their reps and sets.

U.3: Detected error in the user's form during a workout

During the workout phase, the user can monitor their form through the image shown by *Spotter*. No events will occur while the workout is performed with the correct motion and posture. If the user strays from the correct positioning of a specific workout, the application will prompt the user that an error has been made and then suggest the corrections to make.

U.4: Analyzing a previous workout

If the user chooses to replay a previously recorded workout, the application will continue to monitor the form and positioning of the user in the playback. While analyzing the video, workout data is tracked, and errors are caught and shown to the user if there are any. During the playback, the user can monitor his/her form through the tracked key positions which will also be highlighted by the application if errors were made. During the playback, the application will also make suggestions to correct any errors that were caught in the video. This is intended to ensure the application is accessible and usable at any given time and not just when the user is performing his/her workouts.

U.5: Viewing Exercise Description

Spotter will also have the ability to describe and suggest various exercises to help the user achieve predetermined goals. If the user is interested in discovering various new exercises to perform, *Spotter* will provide the user with the required information to correctly perform said exercises. This will include equipment if any, ideal reps/sets count, the area of impact/focus, and the correct form. By providing the user with the information for various exercises, *Spotter* will become the one-stop-shop for learning exercises and performing them correctly.

Context Diagram

The following is a context diagram of the Spotter System:

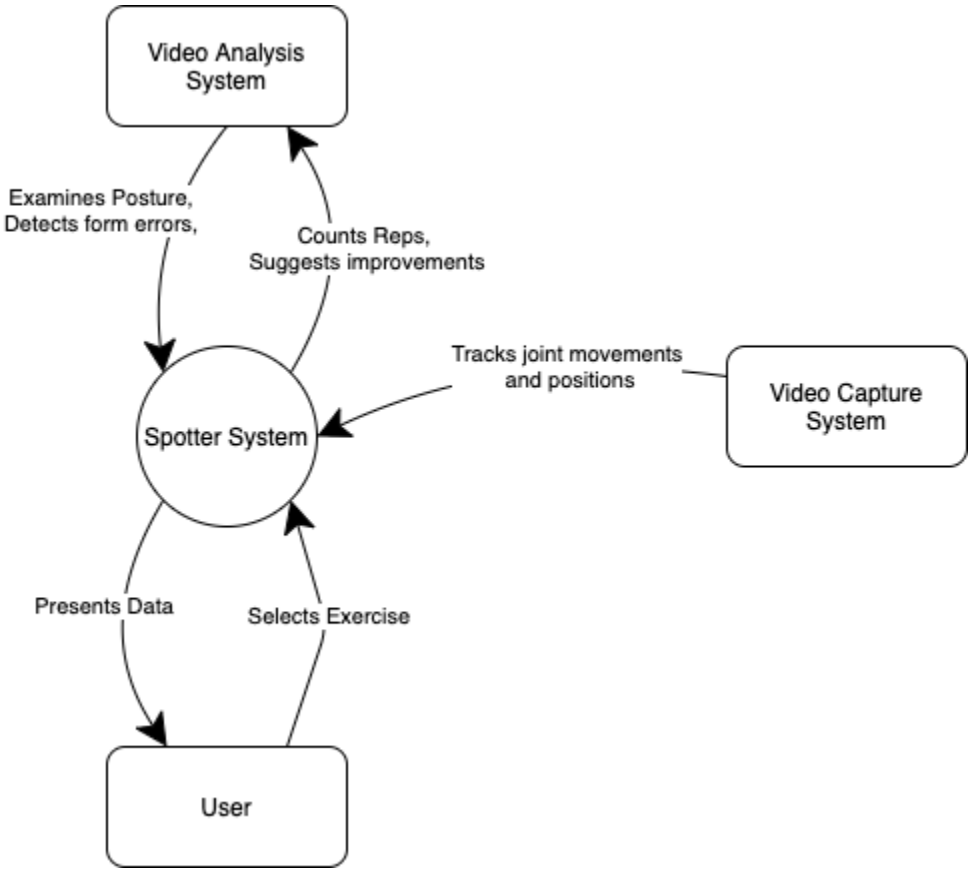

Figure 1. Context diagram showing the interactions between the involved systems

Monitored and Controlled Variables

Monitored Variables

Monitor Name	Monitor Type	Range	Units	Comment(s)
Left shoulder coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of left shoulders detected in a picture / video frame.
Right shoulder coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of right shoulders detected in a picture / video frame.
Left elbow coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of left elbows detected in a picture / video frame.
Right elbow coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of	Coordinates of right elbows detected in a picture / video frame.

			$\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	
Left wrist coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of left wrists detected in a picture / video frame.
Right wrist coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of right wrists detected in a picture / video frame.
Left hip coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of left hips detected in a picture / video frame.
Right hip coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of right hips detected in a picture / video frame.
Left knee coordinates	Position	x: [0, 1] y: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of	Coordinates of left knees detected in a picture / video frame.

		z: [0, 1]	$\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	
Right knee coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of right knees detected in a picture / video frame.
Left ankle coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of left ankles detected in a picture / video frame.
Right ankles coordinates	Position	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	Coordinates of right ankles detected in a picture / video frame.

Controlled Variables

Controlled Name	Controlled Type	Range	Units	Comment(s)
Max refresh rate	Frequency	[5, 100]	$\frac{frames}{second}$	The maximum number of frames per second that the program will track
Sensitivity	Threshold	x: [0, 1] y: [0, 1] z: [0, 1]	x: ratio of $\frac{distance\ away\ from\ left\ of\ the\ scree}{screen\ width}$ y: ratio of $\frac{distance\ away\ from\ bottom\ of\ the\ sc}{screen\ height}$ z: ratio of $\frac{distance\ away\ from\ the\ screen}{maximum\ tracking\ distance}$	The minimum change in distance that a landmark needs to have from a previous frame in order for it to be considered a movement.

Project Constraints

Mandated Constraints:

- The project must be completed within the course of the academic year
    - Rationale: The project must be completed and submitted by the end of the academic year as per project requirements.
- The cost of the project must not exceed $800
    - The project must be economically feasible.

Diagrams Showing Functional Decomposition

User Diagram

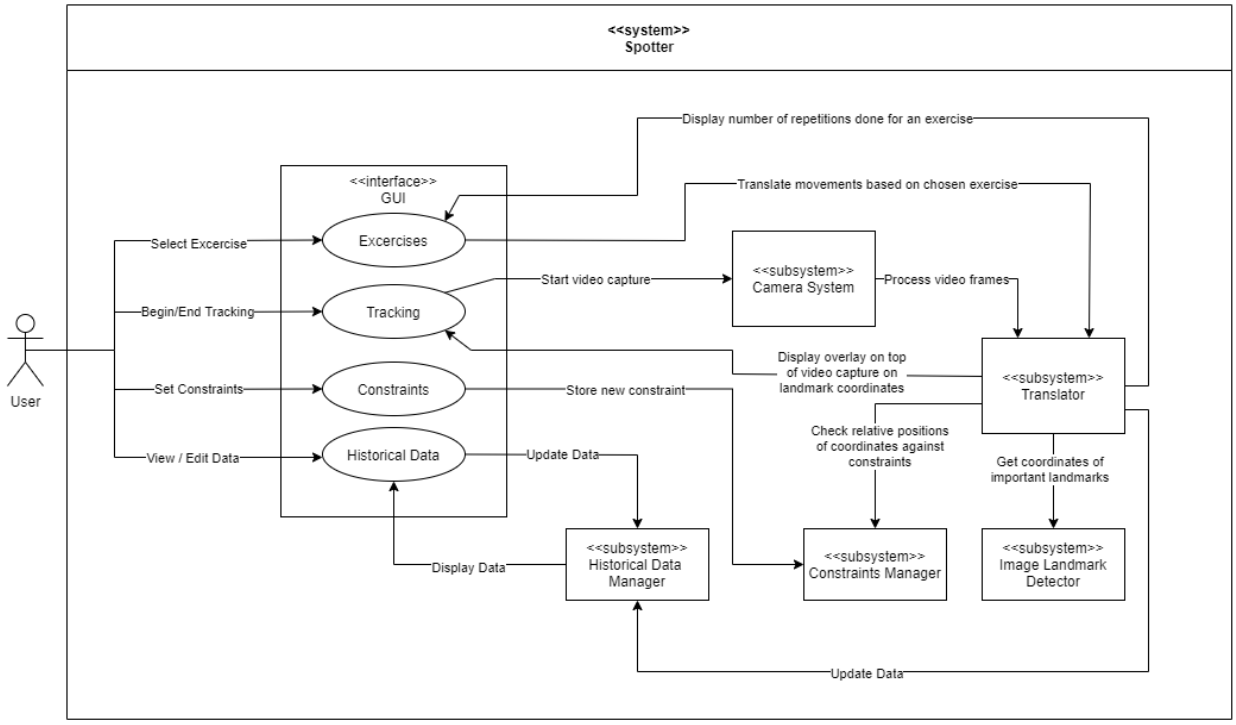

Figure 2. User diagram showing possible interactions between a user and the system.

Data flow diagram

Landmark: Objects within images we are interested in. The **Landmark Detector** finds the location of these landmarks and returns the coordinates when given an image.

Constraints: Constraints on the positions of landmarks relative to other landmarks. These are used to determine whether an exercise is done correctly or not.

Translator: Uses the **Landmark Detector** on incoming video frames to obtain coordinates of **Landmarks** in order to cross reference them against any **Constraints** for a user chosen exercise. This should output data such as number of correct reps done and sections of the video feed where the exercise was not done with correct form in accordance with the constraints.

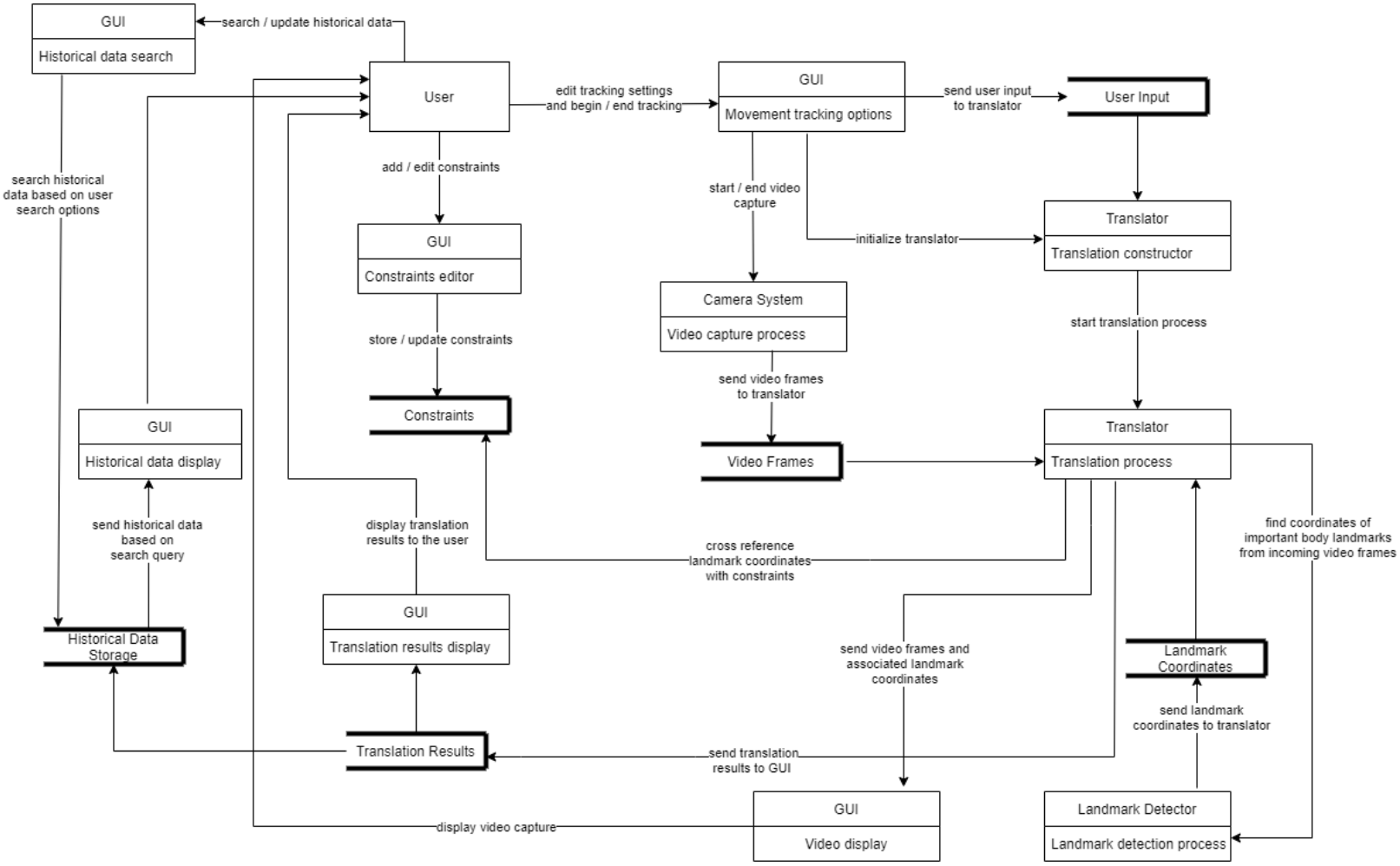

Figure 3: Data Flow Diagram outlining the key components of Spotter.

Non-Functional Requirements

Speed Requirements

NF1	The response time of the app when tracking movement must be less than ten milliseconds.
Rationale	The user must be able to tell if they are doing an exercise wrong in real time.
NF2	The website's load time for each page must be less than one second under normal conditions.
Rationale	Improves user experience due to low load times. Users will be able to tell right away if their internet is the issue.
NF3	The website's interface takes less than ten milliseconds to display any changes made in the backend.
Rationale	Improves user experience due to low load times.

Scalability

NF4	The app must be able to support an annual growth of 15% of new users.
Rationale	The app must be able to scale with demand, otherwise, performance issues will arise.

Reliability

NF5	The app will continue working if one of the features is not working, apart from movement tracking, since this is a critical feature.
Rationale	If one part of the app is not working, the other parts are still working. This is to allow users to still be able to use the app regardless of whether they have or do not have access to certain features.

Availability

NF6	The app shall have a worst case uptime of 98% an expected uptime of 99% and a best case uptime of 99.5%.
Rationale	Different users workout at different times, allowing our app to be available as much as possible allows our users to depend on us when they need to workout.

Regulatory

NF7	The app must meet the WCAG 2.0 Level AA success criteria for accessible websites.
Rationale	This is a requirement by the government of Canada. It is to allow everybody, including those with disabilities, to use our website efficiently.

Recoverability

NF8	In the occurrence of an outage, the app will be able to come back up within less than a minute.
Rationale	We want people to be able to continue their workouts as quickly as possible.

Capacity

NF9	The app must be able to support up to 100,000 concurrent users.
Rationale	This is an expected number of users we will have around 1 year after launching the app.

Maintainability

NF10	The average maintenance time for the app must be less than ten minutes every two weeks.
Rationale	Maintenance times should be kept to a minimum to allow other work to be done on the app.

Serviceability

NF11	The average time to release a fix for a bug must be less than one minute.
Rationale	We want our users to be using a reliable and correct system at all times.

Security

NF12	The app will not store any information pertaining to the user once the user stops using the app.
Rationale	This proves to our users that we do not use or distribute their data to other entities and also provides security for our users since there will be no risk of getting user data stolen by hackers.

Manageability

NF13	The app notifies the administrators when it is down and when the app is experiencing slowness.
Rationale	This allows the administrators to address the issues in a timely manner.

Environmental

NF14	The app must be able to run on the most recent version of any browser, mobile and desktop, and the version before that.
Rationale	This allows most people to use our app.

Data Integrity

NF15	The app will not share the metadata used to connect to the website to any third parties.
Rationale	Provides assurance to users that their data is not being distributed without their consent.
NF16	The app will delete all metadata used to connect to the app once the user leaves the website.
Rationale	The app does not store any user data once the user has stopped using the app, this adds to the users security as well.

Interoperability

NF17	The app uses the user's local device for storage.
Rationale	This is how users will access their data from the app.

Usability

NF18	The website’s interface must be user-friendly and easy to use.

Undesired Event Handling

Event	**How the event is handled**
No webcam detected.	The “begin tracking” option will not begin until the user connects a webcam or enables the camera option in their mobile device.
More than one person detected by the app.	If the tracking system detects more than one person, the app will notify the user of this and the other person must leave the area where the tracking system is tracking.
Webcam is having difficulty detecting a person.	If the tracking system is having difficulty tracking a person, whether it be due to a blurry lens or any other reason, the app will notify the user to double check their webcam so that the tracking system can properly identify the user.
There is no more storage on a user's local device.	This will be handled by the operating system in which the user is using this app. The app will notify the user that the data could not be stored.
The user is unable to connect to the website.	If the website is under maintenance, the website will load a banner saying that the website is currently under maintenance. If everything is ok on the server side, the user must check their internet connection.

List of Requirements that are not Likely to Change

NF2

Key contributor to user engagement.

NF5

Key product feature.

NF7

Legally required if an organization consists of 50 or more employees.

NF10

Key contributor to user experience.

NF11

Key contributor to user experience.

NF13

Key to ensuring product uptime.

NF14

Key to reaching and retaining users.

NF15

Need to share metadata with third parties not anticipated.

NF16

Need to store metadata not anticipated.

NF17

Key product feature.

NF18

Key to user experience.

List of Requirements that are Likely to Change

NF1

A time constraint of more than 10 milliseconds may be acceptable as long as it does not noticeably impact user experience.

NF3

A time constraint of more than 10 milliseconds may be acceptable as long as it does not noticeably impact user experience.

NF4

Subject to change due to consumer demand.

NF6

Subject to change due to unforeseeable events.

NF8

Time constraints may change due to user feedback.

NF9

Subject to change due to consumer demand.

NF12

May change depending on implementation as long as security is not compromised.