***Get Fit Right Project Report***



**Prepared by:**

**Group 12**

**Ahmed Khan, Edgar Martinez, Rameez Baig, Zakee Jabbar**

**CS 440**

**at the**

**University of Illinois Chicago**

**Fall 2018**

**Table of Contents**

[List of Figures 7](#_Toc530938187)

[List of Tables 8](#_Toc530938188)

[I Project Description 9](#_Toc530938189)

[1 Project Overview 9](#_Toc530938190)

[2 The Purpose of the Project 9](#_Toc530938191)

[2a The User Business or Background of the Project Effort 9](#_Toc530938192)

[2b Goals of the Project 10](#_Toc530938193)

[2c Measurement 10](#_Toc530938194)

[3 The Scope of the Work 10](#_Toc530938195)

[3a The Current Situation 10](#_Toc530938196)

[3b The Context of the Work 11](#_Toc530938197)

[3c Work Partitioning 11](#_Toc530938198)

[3d Competing Products 12](#_Toc530938199)

[4 The Scope of the Product 12](#_Toc530938200)

[4a Scenario Diagram(s) 13](#_Toc530938201)

[4b Product Scenario List 13](#_Toc530938202)

[4c Individual Product Scenarios 13](#_Toc530938203)

[5 Stakeholders 14](#_Toc530938204)

[5a The Client 14](#_Toc530938205)

[5b The Customer 14](#_Toc530938206)

[5c Hands-On Users of the Product 14](#_Toc530938207)

[5d Maintenance Users and Service Technicians 15](#_Toc530938208)

[5e Other Stakeholders 15](#_Toc530938209)

[5f User Participation 16](#_Toc530938210)

[5g Priorities Assigned to Users*.* 16](#_Toc530938211)

[6 Mandated Constraints 16](#_Toc530938212)

[6a Solution Constraints 16](#_Toc530938213)

[6b Implementation Environment of the Current System 17](#_Toc530938214)

[6c Partner or Collaborative Applications 17](#_Toc530938215)

[6d Off-the-Shelf Software 18](#_Toc530938216)

[6e Anticipated Workplace Environment 18](#_Toc530938217)

[6f Schedule Constraints 19](#_Toc530938218)

[6g Budget Constraints 19](#_Toc530938219)

[7 Naming Conventions and Definitions 19](#_Toc530938220)

[7a Definitions of Key Terms 19](#_Toc530938221)

[7b UML and Other Notation Used in This Document 20](#_Toc530938222)

[7c Data Dictionary for Any Included Models 20](#_Toc530938223)

[8 Relevant Facts and Assumptions 20](#_Toc530938224)

[8a Facts 20](#_Toc530938225)

[8b Assumptions 21](#_Toc530938226)

[II Requirements 22](#_Toc530938227)

[9 Product Use Cases 22](#_Toc530938228)

[9a Use Case Diagrams 22](#_Toc530938229)

[9b Product Use Case List 22](#_Toc530938230)

[9c Individual Product Use Cases 23](#_Toc530938231)

[10 Functional Requirements 25](#_Toc530938232)

[11 Data Requirements 28](#_Toc530938233)

[12 Performance Requirements 29](#_Toc530938234)

[12a Speed and Latency Requirements 29](#_Toc530938235)

[12b Precision or Accuracy Requirements 30](#_Toc530938236)

[12c Capacity Requirements 30](#_Toc530938237)

[13 Dependability Requirements 31](#_Toc530938238)

[13a Reliability Requirements 31](#_Toc530938239)

[13b Availability Requirements 32](#_Toc530938240)

[13c Robustness or Fault-Tolerance Requirements 32](#_Toc530938241)

[13d Safety-Critical Requirements 32](#_Toc530938242)

[14 Maintainability and Supportability Requirements 33](#_Toc530938243)

[14a Maintenance Requirements 33](#_Toc530938244)

[14b Supportability Requirements 33](#_Toc530938245)

[14c Adaptability Requirements 34](#_Toc530938246)

[14d Scalability or Extensibility Requirements 35](#_Toc530938247)

[14e Longevity Requirements 35](#_Toc530938248)

[15 Security Requirements 36](#_Toc530938249)

[15a Access Requirements 36](#_Toc530938250)

[15b Integrity Requirements 36](#_Toc530938251)

[15c Privacy Requirements 37](#_Toc530938252)

[15d Audit Requirements 37](#_Toc530938253)

[15e Immunity Requirements 37](#_Toc530938254)

[16 Usability and Humanity Requirements 37](#_Toc530938255)

[16a Ease of Use Requirements 37](#_Toc530938256)

[16b Personalization and Internationalization Requirements*.* 38](#_Toc530938257)

[16c Learning Requirements 39](#_Toc530938258)

[16d Understandability and Politeness Requirements 39](#_Toc530938259)

[16e Accessibility Requirements 39](#_Toc530938260)

[16f User Documentation Requirements 40](#_Toc530938261)

[16g Training Requirements 41](#_Toc530938262)

[17 Look and Feel Requirements 41](#_Toc530938263)

[17a Appearance Requirements 41](#_Toc530938264)

[17b Style Requirements 42](#_Toc530938265)

[18 Operational and Environmental Requirements 42](#_Toc530938266)

[18a Expected Physical Environment 42](#_Toc530938267)

[18b Requirements for Interfacing with Adjacent Systems 42](#_Toc530938268)

[18c Productization Requirements 43](#_Toc530938269)

[18d Release Requirements 43](#_Toc530938270)

[19 Cultural and Political Requirements 43](#_Toc530938271)

[19a Cultural Requirements 43](#_Toc530938272)

[19b Political Requirements 44](#_Toc530938273)

[20 Legal Requirements 46](#_Toc530938274)

[20a Compliance Requirements 46](#_Toc530938275)

[20b Standards Requirements 46](#_Toc530938276)

[21 Requirements Acceptance Tests 47](#_Toc530938277)

[21a Requirements – Test Correspondence Summary 47](#_Toc530938278)

[21b Acceptance Test Descriptions*.* 47](#_Toc530938279)

[III Design 51](#_Toc530938280)

[22 Design Goals 51](#_Toc530938281)

[23 Current System Design 51](#_Toc530938282)

[24 Proposed System Design 51](#_Toc530938283)

[24a Initial System Analysis and Class Identification 51](#_Toc530938284)

[24b Dynamic Modelling of Use-Cases 51](#_Toc530938285)

[24c Proposed System Architecture 52](#_Toc530938286)

[24d Initial Subsystem Decomposition 52](#_Toc530938287)

[25 Additional Design Considerations 52](#_Toc530938288)

[25a Hardware / Software Mapping 52](#_Toc530938289)

[25b Persistent Data Management 52](#_Toc530938290)

[25c Access Control and Security 52](#_Toc530938291)

[25d Global Software Control 53](#_Toc530938292)

[25e Boundary Conditions 53](#_Toc530938293)

[25f User Interface 53](#_Toc530938294)

[25g Application of Design Patterns 53](#_Toc530938295)

[26 Final System Design 53](#_Toc530938296)

[27 Object Design 53](#_Toc530938297)

[27a Packages 54](#_Toc530938298)

[27b Subsystem I 54](#_Toc530938299)

[27c Subsystem II 54](#_Toc530938300)

[27d etc. 54](#_Toc530938301)

[IV Project Issues 54](#_Toc530938302)

[28 Open Issues 54](#_Toc530938303)

[29 Off-the-Shelf Solutions 54](#_Toc530938304)

[29a Ready-Made Products 55](#_Toc530938305)

[29b Reusable Components 55](#_Toc530938306)

[29c Products That Can Be Copied 55](#_Toc530938307)

[30 New Problems 55](#_Toc530938308)

[30a Effects on the Current Environment 55](#_Toc530938309)

[30b Effects on the Installed Systems 55](#_Toc530938310)

[30c Potential User Problems 55](#_Toc530938311)

[30d Limitations in the Anticipated Implementation Environment That May Inhibit the New Product 56](#_Toc530938312)

[30e Follow-Up Problems 56](#_Toc530938313)

[31 Migration to the New Product 56](#_Toc530938314)

[31a Requirements for Migration to the New Product 56](#_Toc530938315)

[31b Data That Has to Be Modified or Translated for the New System 56](#_Toc530938316)

[32 Risks 56](#_Toc530938317)

[33 Costs 56](#_Toc530938318)

[34 Waiting Room 57](#_Toc530938319)

[35 Ideas for Solutions 57](#_Toc530938320)

[36 Project Retrospective 57](#_Toc530938321)

[V Glossary 57](#_Toc530938322)

[VI References / Bibliography 57](#_Toc530938323)

[VII Index 58](#_Toc530938324)

### ****List of Figures****

[Figure 1 – Context of Work……………………………………………………………………...11](#_The_Context_of)

[Figure 2 - Sample Use Case Diagram from Bruegge & DuToit ( modified ) 21](#_Toc525544068)

[Figure 3 - Sample Use Case Diagram from Robertson and Robertson 22](#_Toc525544069)

### ****List of Tables****

[*Table 2 - Requirements - Acceptance Tests Correspondence* 61](#_Toc525544085)

# Project Description

The *Get Fit Right* is a technology which utilizes motion capture to analyze the movements of a user while performing an exercise. *Get Fit Right* has two modes. A smart phone only mode which relies on the phones camera to track the user and give feedback. The second mode utilizes special body suits with built in sensors to connect to the phone and give more accurate data regarding the user. The second mode uses the data from the body suit and the smart phone to provide feedback. If a user is only lacking form, then the application will be able to advise them on what they’re doing wrong. If the error lies in a stiffness within the body, or lack of mobility, then the application will also be able to spot that and advise the user a plan to increase mobility in the problem area. The purpose of the device in general is to provide feedback to a user on the based on their movement to improve performance, increase mobility, and keep them safe during the duration of the exercise.

## Project Overview

The use of improper form during many of the more complex exercises – Deadlift, Squat, Snatch – can lead to injury. The goal of our simulator is taking a user through the exercise in a safe, comfortable environment where the risk of injury is as minimized as possible.

## The Purpose of the Project

This project is being done to combat a variety of problems, such as injury prevention, and to improve performance.

### The User Business or Background of the Project Effort

We are working with Nike, which is an athletic apparel and equipment business, in order to develop the Get Fit Right application. The application suits, which users will be able to wear, will be developed by them as well. The application was made with the intention of mitigating the amount of injury created in the gym. Nike sells products which are meant to improve an athlete’s performance. The creation of this application will allow Nike to sell shirts and pants that are meant to be captured by our motion capture software. The user will be able to utilize that product in the gym and receive accurate useful feedback on their weightlifting activity and improve because of it. Not only do we aim to battle injury and improve performance, but we wish to make an application readily to anyone who wishes to improve their technique. Therefore, we made the motion capture technology to work with just an ordinary smartphone. There is a plethora of terrible advice floating around in the wellness community, and most everyone knows of someone (or is that someone) who deals with gym anxiety because they don’t know if they’re doing something right. For this reason, we wanted to put that power in the hand of our users, and make precise, useful information readily available to anyone who wants to make a difference in their health.

### Goals of the Project

Our client, Nike, has just as much to benefit from their involvement in our technology as the users will. Our technology utilizes a smartphone for our motion capture technology, however, it also utilizes specialized shirts and pants that are created by our Client. The cost of a new workout outfit is far less than that of a personal trainer. The product is being made in order be the premiere choice for a user of any kind looking to improve their physique, or performance. We want to be able to give instant feedback to any user who utilizes our application. It should be intuitive, easy to use, and provide real usable advice.

### Measurement

Our app will have the ability for users to enter the height, weight, and test for flexibility. Upon the first use of our application, the Get Fit Right will administer a test in order to track where our users starting point is. Upon utilizing the application, one will be able to check in on their progress since every time they use it, data is collected. Improvements are measured and displayed in an easy depict visual representation. Once users see the results, we will be able to measure the success of the apps by the frequency that they use it, and by the reviews they leave on the IOS/Android respective app stores.

## The Scope of the Work

An application that allows people with any type of workout experience to correctly and safely workout by notifying them when performing an exercise incorrectly.

### The Current Situation

There are already lots of fitness apps out there for people to use to help them get fit and stay healthy. From running apps that track your time per mile to apps that track how much water you drink or food you eat. The current issue with this apps is that they don’t alert the user when they are working out incorrectly, leading to unsafe workouts, resulting in unwarranted injuries when working out.

Our solution consists of an application that uses a client’s camera to track posture when working out and alerting them when their posture is incorrect thus avoiding unsafe workouts. By improving user experience when working out it allows our clients to work out without worrying about technique and focus on what really matters about working out, getting healthy. 3b

### The Context of the Work

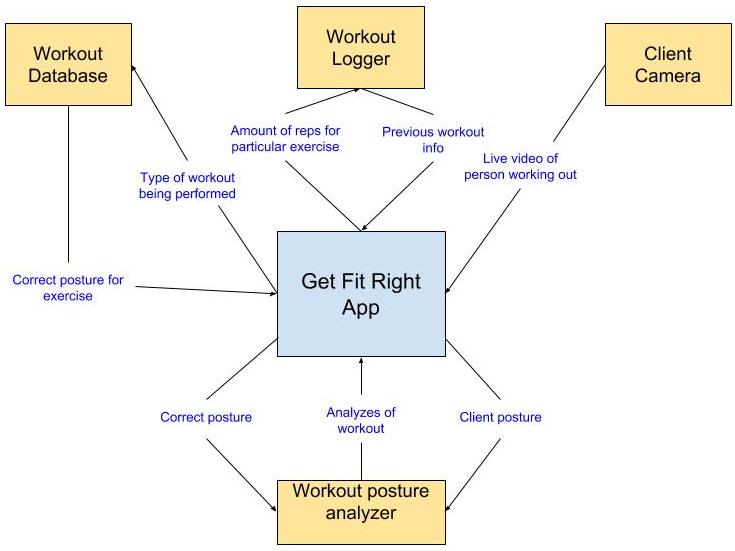
[](#_List_of_Figures)

Figure 1 - Context of Work

### Work Partitioning

|  |  |  |
| --- | --- | --- |
| **Event Name** | **Input and Output** | **Summary** |
| Workout database determining type of workout | Type of workout (out) | Sends the type of workout so to be able to receive the posture of the exercise. |
| Workout database knowing the workout | Posture Info (in) | Sends the information for the type of posture needed for that exercise. |
| Client camera video | Live video of workout (in) | Video received from camera that will be used to determine if posture is correct. |
| Posture video analyzer | Posture info and live video of workout (out) | Receives both the correct posture info and the live video of the workout that will be compared to inform the user if his/her posture is correct. |
| Posture video analyzer result | Update of workout (in) | Update to the app that will notify the user if posture is correct in live time based on results from the analyzer. |
| Exercise logger | Exercise Info (out) | Sends the amount of exercise and other data gathered the user did to cloud storage system so that user can access data from any of his/her devices. |
| Exercise history | Exercise Info (In) | Retrieves the previous workout sessions info so that user sees progression over time. |

### Competing Products

There are currently many applications available to help people get into shape and teach them new exercises, but none of them notify the user if they are correctly performing their workout. The current selection of applications offer the user no notifications in real time to inform them if they are correctly performing the exercise, thus putting people in danger of needless injuries. Most of these applications just offer videos or images of the workout to the user, completely ignoring people’s safety when performing new exercises by offering live feedback to the user based on their posture.

## The Scope of the Product

The main idea behind this product is to be able to tell a user if the workout that he or she is performing is being done correctly. This is done using a person’s smartphone and capturing video of them working out and instantly notifying them if they are doing their workout using good or bad technique. Also included in the application will be a workout logger that helps the user lookup previous workout and what type of exercises were performed that day and other valuable information like the amount of repetitions and if the workout was done using the correct posture and technique.

### Scenario Diagram(s)

A close up of text on a white background

Description generated with high confidence

Figure 2 - Project Scenario

### Product Scenario List

1. Initial Startup
2. Starting a workout
   1. Using Smartphone camera
   2. Live feedback of workout
3. Looking at previous workouts

### Individual Product Scenarios

1. **Initial Startup:** John decides to run Get Fit Right app on his smartphone. He has received as a gift for his birthday a set of weights, problem is that he doesn’t know if he is correctly performing his workouts. So when he opens the application for the first time John is asked to input his current vitals and measurements. After he enters his information he is given the option to start a workout or to look at previous workouts.
2. **Starting a workout:** John decides to start a workout so on his smartphone he selects this option. By doing so John will receive notifications on his phones screen in real time as he works out displaying whether he is correctly performing the workout based on his posture and comparing it to the correct posture. John also places his phone in at an angle that records his workout for the analytical part.
   1. **Using Camera:** By using his camera John is recording his workout and its being analyzed in real time using machine learning on the local device and it will notify him if the workout is done correctly.
   2. **Receiving Feedback:** John has completed a set of repetitions for a barbell squat. He looks at his phone and proceeds to hit the “DONE” button. The app then takes a second to process all of the data, and then displays a list of analyses of John’s set. After John sees the analyses, the app then creates recommendations based on John’s performance. The app recognizes that John’s form quickly turned bad after rep 4 out of 8, therefore the app recommends John to use a lighter weight for the next set in order to avoid injury. John presses “CONTINUE” on the app and moves on to the next set.
3. **Looking at previous workouts:** John doesn’t feel like working out today, but instead wants to reminisce about last Tuesday’s workout. He wants to see how many squats he performed. Along with how many squats he did John can also see how much weight he squatted that day along with data that shows if John performed his workout correctly based on the analysis that was done that day.
   1. **Retrieving Data:** John decides to use his computer one day because his phone broke to view last weeks workout data and is able to because his data was sent to the cloud so that he can access it from any of his devices and doesn’t have to worry about it getting lost which makes John very happy since he can show his workouts to other people from anywhere in the world on any device connected to the internet.

## Stakeholders

### The Client

The client is Nike Incorporated, a multinational corporation that is a leader in the design, development, and manufacturing of footwear, apparel, equipment, accessories, and services.

### The Customer

The intended purchasers of this product are professional sports teams, gym owners, and personal trainers. Any individual who helps others improve their fitness is a potential customer.

### Hands-On Users of the Product

Professional Athletes: Athletes will use this app to improve their overall performance. From tracking a throw from the quarterback to the swing of a power hitter, the app will provide feedback on the essential skillsets of an athlete in any sport. Professional athletes are already at a master level in their fields, but this will allow them to further practice their “textbook” skills, so that they can make “textbook” plays. Athletes already use high tech technology for their training, and this will be a perfect add-on to their training options. Since their technological experience is already at a master level, using any kind of new technology will be a breeze. Apart from improving their in-game skills, athletes can also use this to assist with their weight training and agility workouts.

Gym Goers: There are varying levels of gym goers. There are fitness buffs, who walk around the gym with their duffel bags and their protein shakes to take before, during, and after their workouts. Their knowledge of fitness and exercise ranges from a journeyman to a master level. These types of people are looking for that something extra to make them lose that one percent of body fat or make them hit their next PR (persona record). The other subgroup are the regular gym members. Their knowledge of fitness and exercise ranges from novice to journeyman. Gym goers would use this app to ensure they are performing the exercises with proper form and repetitions. The technological experience of this group varies from novice to master. The older gym goers tend to be novices, while the younger gym goers tend to be masters.

### Maintenance Users and Service Technicians

There would be a team of software developers that maintain and update the app, add new features, and resolve bugs that are discovered by the users. Another team would be required to service the hardware aspects of this product. The hardware team would make changes and create new designs of the clothing pieces, as well as troubleshoot any issues that arise in either connectivity or usability of the product. Ideally, there would be an employee at every Nike store who was trained to help with issues regarding this product.

### Other Stakeholders

Patent lawyers: Lawyers will ensure that others do not steal our ideas and products. They will follow all rules and regulations and prevent the company from intellectual property infringement.

Fitness Professionals: Provide data for our exercise database. Give us baselines on how to perform a certain exercise. These can include coaches, trainers, doctors, and scientists.

Google Fit, MyFitnessPal: Current popular fitness apps that would need to add new features to keep up with *Get Fit Right*. These apps only cover limited amount of our features and tend to serve the same audience.

Adidas, Under Armour: These companies would need to create a similar app to keep up with Nike which is one of their main competitors. Adidas recently lost its NBA contracts to Nike. To avoid further losses in business these companies would need to invent new products.

### User Participation

Professional Athletes: Athletes can suggest different actions and maneuvers that they would like special training on. Preferably specific skills that can’t be assessed manually.

Gym Goers: Gym goers can test out the preliminary versions of the UI to give feedback and suggestions to the developers to create a better UI for when the product launches.

### Priorities Assigned to Users*.*

Key Users: Gym Goers (80% of users) – Gym goers will be the primary users of this app. These are the users that need the most feedback on their exercises as a lot of these users are in the novice to journeyman level of expertise on the topic.

Secondary Users: Professional Athletes (20% of users) – Athletes don’t require the same level of assistance and feedback as gym goers do. Their use for this app is more of an additional tool to improve and track their performance. This app won’t be their primary tool for improvement, so it is better to spend more time developing towards the gym goers.

## Mandated Constraints

This section describes constraints on the design of the product. These constraints have been mandated by the client at the beginning of the project before the full set of requirements had been determined and are set in stone. Any other solution that does not follow these constraints are unacceptable.

### Solution Constraints

Description: The software of the product shall be available on the Apple App Store and the Google Play Store.

Rationale: The client wants as many customers as possible to be able to use the base version of the application without having to purchase additional hardware.

Fit Criterion: The product shall be compliant with Apple and Google guidelines and requirements in order to be available on each respective app store.

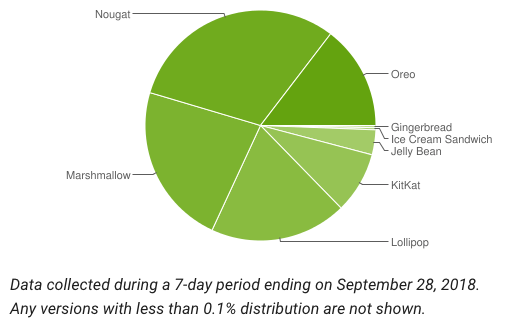
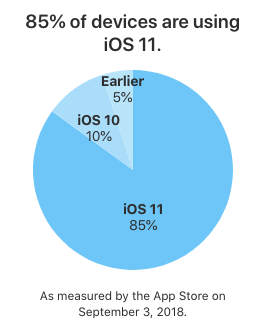
Description: The software of the product shall be downloadable on a minimum of 85% devices for each respective app store.

Rationale: The client does not want to limit its user base simply on what version of iOS/Android OS the customer’s device is running.

Fit Criterion: Percentage distribution of devices and operating system versions can be found on Google’s and Apple’s distribution dashboards:

Apple: <https://developer.apple.com/support/app-store/>

Google: <https://developer.android.com/about/dashboards/>

**

Description: The wearable technology available in the advanced system must be compliant with United States clothing guidelines and regulations.

Rationale: The advanced system of the product will only be sold within the United States; therefore, it must follow guidelines set by the government.

Fit Criterion: Must follow all regulatory guidelines through the Federal Trade Commission under the Textile and Wool Acts.

### Implementation Environment of the Current System

The iOS version of the application must be developed using Swift and not Objective-C.

Swift is given preference over Objective-C due to the fact that the latter is an outdated technology that is more is more prone to error than the former.

The Android version of the application must be developed using Java and not Kotlin.

Java is given preference over Kotlin due to the fact that the former is a very well-known language throughout the world as compared to the latter. According to the client, Kotlin is too young of a language to be used in an enterprise application.

### Partner or Collaborative Applications

External applications with which this product must be compatible with include the Apple Health app and the Google Fit app.

These are the applications that are maintained by the owners of the operating system the product will be supported on, therefore there is a greater chance that the user utilizes these applications.

The product must be able to read in user data from these apps, but also be able to update the data by writing to these apps. This has been mandated to allow the user to easily monitor and track his/her progress in parallel with any other third-party application that is also compatible with Apple Health and Google Fit.

### Off-the-Shelf Software

The product must implement Adobe Acrobat Reader.

This is required to print an editable PDF of the user’s data and progress through the most commonly accepted means of distribution.

Adobe Acrobat Reader is a free software tool used to read PDF documents. Files are handled differently depending on the operating system, and the client does not want to alienate a certain population of customers simply on the operating system he/she prefers. This also allows the customer to share necessary data easily with his/her trainer or primary care physician without worry since PDFs and Adobe Acrobat Reader are not reliant on a license, like Microsoft Word and Excel are.

### Anticipated Workplace Environment

The general description of the workplace is an open floor environment that allows the user to easily and freely perform the desired exercise within adequate range of the sensory camera. Environments include, but are not limited to recreational and professional gyms, physical therapy offices, and any environment that is deemed safe for the pursuant activity.

The workplace is noisy; therefore, notifications must be visible and audible at a high enough frequency to be seen and heard.

The workplace is large; therefore, the application must be able to accurately measure distance from the user.

The workplace is potentially dangerous; therefore, the application must not distract the user from being aware of his/her surroundings.

The workplace is fast paced; therefore, the application must be able to track and distinguish the user and his/her equipment from others.



### Schedule Constraints

A working demo of the application must be presented to the client every 3 months to show what progress has been made, and whether certain functionality must take precedence over others.

The final release of the applications on the App/Play Stores and the release of the apparel has been planned for the end of 2020 to meet the needs of the client.

Once the apps have been released, frequent updates to the applications must take place to fix bugs and optimize the applications on future operating system upgrades.

### Budget Constraints

The client has proposed $100,000 to successfully create the applications for iOS and Android, and another $100,000 to successfully create the apparel that will be used for the advanced system of the application.

Based off estimates from <http://howmuchtomakeanapp.com/> this budget is very realistic to build this product.

Factors that have been considered for the budget constraints include developer and designer salaries, infrastructure costs for the machine learning database, and warehouse rent and employee maintenance for the apparel.

## Naming Conventions and Definitions

### Definitions of Key Terms

Below is the list of some of the most important words which include acronyms and abbreviations. See the glossary at the end of the document for a complete list.

“Textbook” – When something is executed perfectly it can be described using this word.

PR – Personal records. Common uses is when an individual reaches a new weight on a lift or a runner completes a race beating their own personal record.

Other important keywords will be added as necessary.

### UML and Other Notation Used in This Document

This document generally follows the Version 2.0 OMG UML standard, as described by Fowler in [1]. Any exceptions are noted where used.

### Data Dictionary for Any Included Models

All data would be stored in tables in the back-end database.

Table 1: User Profile Data

Fields: Name, Age, Sex, Date of Birth, Weight, Height. Activity Level (On a scale of 1-4 how active is a user)

Table 2: User History Data

Fields: Activity, Duration, Reps (If applicable), Sets (If applicable), Accuracy (Percentage of how accurate the activity of performed), Suggestions (Future actions suggested to the user)

Table 3: Exercise Data

Fields: Name (Name of exercise), Category (Under what general category does this exercise lie), User type (Which user is this meant for), Suggestions (List of suggestions for this exercise), Accuracy Info (Evaluation metrics for this exercise)

Other tables will be added as necessary.

## Relevant Facts and Assumptions

### Facts

iOS code written in Swift must be in Xcode and Xcode is only available on Mac OS.

Android code written in Java must be in Android Studio and Android Studio is available on all platforms.

Android applications use Material Design, while iOS applications use Human Interface Design.

Every smart phone camera has a different number of front-facing megapixels. smart

Men and women sizes are labeled differently.

### Assumptions

Cotton is the best well-rounded material to use when making clothes.

The user already owns an iOS or an Android OS device.

Certain open-source machine learning APIs are already available to integrate and expand upon within this application

User data is readily available through the Apple Health app or the Google Fit app.

Monetary resource disbursements are released at the end of every demo update with the client.

The product cannot move on to the next stage after a demo update if functionality or aesthetic implementation does not please the client.

# Requirements

## Product Use Cases

### Use Case Diagrams

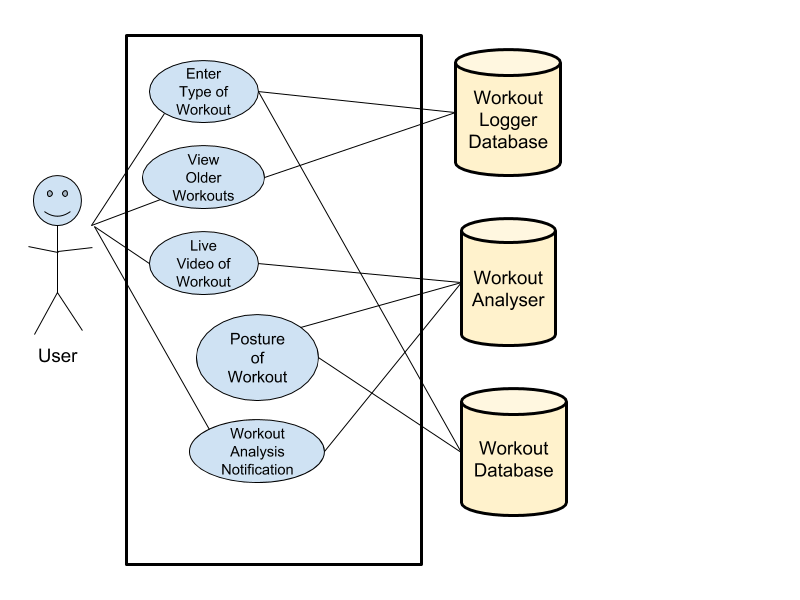


Figure 3 - Use Case Diagram

### Product Use Case List

Enter Workout

View Older Workouts

Live Video of Workout

Posture of Workout

Workout Analysis Notification

### Individual Product Use Cases

|  |  |
| --- | --- |
| *Use Case Name & ID:* | Enter Workout, #1 |
| *Participating Actors:* | Initiated by User  Communicates with workout logger database and workout database |
| *Flow of Events:* | 1. User Enter workout wished to be performed 2. Connects with logger database to save the workout performed 3. Connects with workout database to load posture data into the analyzer |
| *Pre-Conditions:* | Databases are connected to users account based on app sign-in |
| *Post conditions:* | Both databases connect and have workout available for user to enter |

-------------------------------------------------------------------------------------------------------------------------------------------------------------------

|  |  |
| --- | --- |
| *Use Case Name & ID:* | View Older Workouts, #2 |
| *Participating Actors:* | Initiated by User  Communicates with workout logger database |
| *Flow of Events:* | 1. User Enter workout wished to be viewed from previous workout sessions 2. Connects with logger database to load the previous workout performed and loads data for user to see on screen 3. Shows percent of workout performed with correct posture 4. Shows number of repetitions and sets along with the weight performed 5. Ranking the user gave towards the intensity of the exercise |
| *Pre-Conditions:* | Database is connected to users account based on app sign-in process |
| *Post conditions:* | Databases connect and have workout information for user to see |
| *Quality Requirements:* | Data must be fetched within 3 seconds so that user doesn’t have to wait to see previous workouts |

------------------------------------------------------------------------------------------------------------------------------------------------------------------

|  |  |
| --- | --- |
| *Use Case Name & ID:* | Live Video of Workout , #3 |
| *Participating Actors:* | Initiated by application  Communicates with workout analyzer |
| *Flow of Events:* | 1. User places camera to point to him/her 2. Users phone then displays camera view on screen so that user can see himself/herself |
| *Pre-Conditions:* | User has selected an exercise that is available for analysis on the database |
| *Post conditions:* | User can see himself/herself on screen |

------------------------------------------------------------------------------------------------------------------------------------------------------------------

|  |  |
| --- | --- |
| *Use Case Name & ID:* | Posture of Workouts, #4 |
| *Participating Actors:* | Initiated by Time  Communicates with user’s device live video, workout database, and the workout analyzer |
| *Flow of Events:* | 1. Obtain correct workout posture data points from the workout database 2. Use workout analyzer to compare users posture to that of correct workout posture that was obtained from workout database 3. Compare live video posture to that of the database for live analysis |
| *Pre-Conditions:* | User has selected an exercise that is available for analysis on the database |
| *Post conditions:* | Workout analysis is able to show if workout is being performed correctly |
| *Exceptions:* | Users workout that’s selected can’t be analyzed unless correct posture data is available in the workout database |

------------------------------------------------------------------------------------------------------------------------------------------------------------------

|  |  |
| --- | --- |
| *Use Case Name & ID:* | Workout Analyzes Notification, #5 |
| *Participating Actors:* | Initiated by User  Communicates with user’s device and the workout analyzer |
| *Flow of Events:* | 1. Once data is compared by the workout analyzer it sends back to the users device the correctness of the workout 2. Users device then displays that information in real time to inform user of how correctly they are performing their exercise. |
| *Pre-Conditions:* | User has selected an exercise that is available for analysis on the database and live video of user’s exercise is provided as well |
| *Post conditions:* | Users device can show the results as the workout progresses |
| *Exceptions:* | Users workout that’s selected can’t be analyzed unless correct posture data is available in the workout database |

## Functional Requirements

**ID: F1 - Sign into Application**

**Description:**

On initial launch of applications, the user will be asked to sign into his/her account. If the user has no account, he/she will be prompted to make one. If he/she does have an account, he/she will simply sign in.

**Rationale:**

We want to allow our users to be able to save workouts as they are performed and also to be able to view/track old workouts.

**Fit Criterion:**

The system should know if the user has already made an account based upon their email and should ask them to enter their password or if they have forgotten their password. The users account should be linked to them through their email.

**Acceptance Tests:** Test #1

**ID: F2 - Enter workout**

**Description:**

Once a user enters application and is signed into account, he/she will be prompted to select the type of workout, or exercise, they will be performing.

**Rationale:**

We want to allow users to see if they are performing the exercise correct and therefore, they need to select their workout so that our system can analyze it and compare his/her posture to that of correct posture that is available in the workout database.

**Fit Criterion:**

The user’s workout selection should be displayed throughout the workout to make sure that the user is aware of his/her selection. The product will then analyze the workout based upon that user’s selection and make sure that the user is performing the workout correctly.

**Acceptance Tests:** Test #2

**ID: F3 - Obtain Workout**

**Description:**

User will be given the option to load previous workouts from the workout database into their phone.

**Rationale:**

We want to allow our users to be able to see their workout progression over time. For example, the increase in reps or increase in weights used in the exercise.

**Fit Criterion:**

The user should only see data that corresponds to his or her workout and not that of other people. The data should be displayed in a matter that is easily readable by the user.

**Acceptance Tests:** Test #3

**ID: F4 - Live Video of Workout**

**Description:**

After the user selects workout the phone should notify the user that he/she needs to place the phone in a position such that the camera is able to track his movement, more importantly his posture which will be used to know if the user is performing the exercise correctly.

**Rationale:**

To be able to complete exercise posture analysis we want to be able to see the user working out and what better way than using the user’s phones camera to capture that video, so that it can be compared to correct posture of that exercise.

**Fit Criterion:**

The user should be able to easily place his/her device in such a manner that the phones camera is able to capture video while performing the workout.

**Acceptance Tests:** Test #2.

**ID: F5 - Notify User on Workout**

**Description:**

After the user selects workout and positions phone so that camera can record him then he begins his workout. On the user’s device would be a constant notification that allows the user to see if the workout being performed is being done correctly.

**Rationale:**

We want to allow our users to be able to see in real time as their workout is being performed if it’s being done correctly, based upon their posture.

**Fit Criterion:**

The user should be able to easily see on his/her device, while performing the workout, if the workout is being performed correctly based upon the workout analyses.

**Acceptance Tests:** Test #2

## Data Requirements

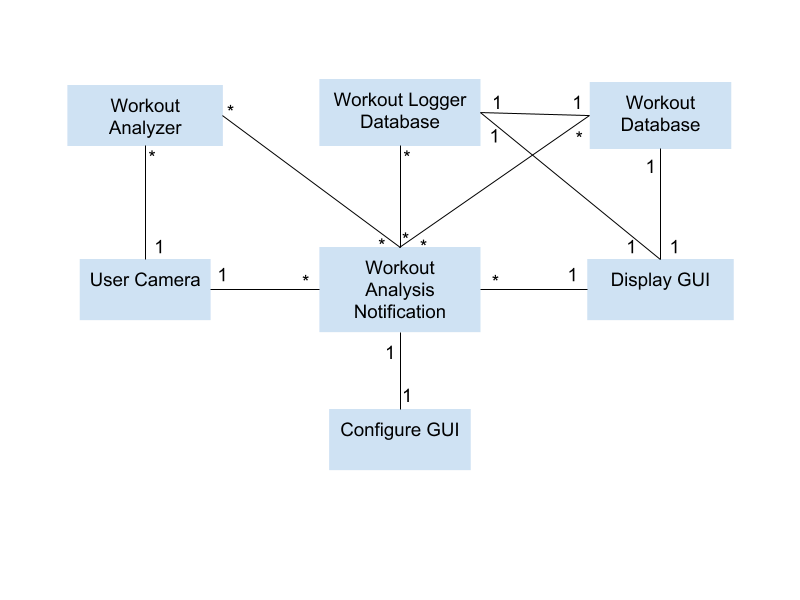


Figure 4 - Data Requirements

**ID: D1 - Obtaining Previous Workouts**

**Description:**

The user should be able to connect to the workout logger database to obtain previous workout sessions.

**Rationale:**

The main benefit of this application is to allow users to see their health progression through time and what a better way than to see how their activates have increased either in doing more sets/reps or gaining better posture by seeing how their posture has improved.

**Fit Criterion:**

The user should see data that corresponds to his or her previous workout. The data should be displayed in a matter that is easily readable by the user.

**Acceptance Tests:** Test #3

## Performance Requirements

### Speed and Latency Requirements

**ID: P1 – Application Sync with Database**

**Description:** The system must refresh all sensory data and sync with the application once per second.

**Rationale:** The application needs to provide immediate feedback to a user based upon his/her performance, therefore the data must be monitored and analyzed constantly.

**Fit Criterion:** The application will receive the current time from the device when the “start exercise” button is pressed and will then maintain the current elapsed time alongside the most recent sync with the machine learning database being used.

**Acceptance Tests:** Test #4

**ID: P2 – Bootup Speed**

**Description:** The system must bootup fast.

**Rationale:** The application is to be used during a user’s workout. Too much time being spent on just loading is time being taken away from the actual workout, and thus can lead to a user giving up on using the application altogether.

**Fit Criterion:** Application should start up in an average speed of 2 seconds 85% of the time.

**Acceptance Tests:** Test #5

**ID: P3 - Quick Video Processing**

**Description:** The system must be able to process a user video for a specific workout quickly.

**Rationale:** The application is to be used during a user’s workout. Too much time being spent on processing the video and giving feedback can lead to a user giving up on using the application altogether. The user would also like quick feedback to avoid consistently doing the workout wrong.

**Fit Criterion:** The application should process a user video and give relevant feedback in less than five seconds on an average basis. The application should show the user a progress bar on how much the video has been processed, so that the user is aware on how much time is left.

**Acceptance Tests:** Test #6

### Precision or Accuracy Requirements

**ID: P4 – Precise Weight Recommendations**

**Description:** Any free weight recommendations should be accurate within five pounds.

**Rationale:** Too much weight in an exercise can lead to injury, and too light of a weight will not be beneficial. Having a maximum difference of five pounds is a light enough weight that will not be too significant but is also heavy enough to produce the desired outcome.

**Fit Criterion:** User feedback of exercises should be satisfied with the recommended weight 90% of the time.

**Acceptance Tests:** Test #7

**ID: P5 - Accurate Repetitions Based on Historical Data**

**Description:** All exercises and repetitions of each exercise should be accurate in accordance to the user’s gender, goals, weight, and age.

**Rationale:** An experienced user can handle a higher intensity workout then someone who is just starting out.

**Fit Criterion:** 95% of users should report that his/her workout was challenging but manageable.

**Acceptance Tests:** Test #8

### Capacity Requirements

**ID: P6 – Priority to Current Active Users**

**Description:** The system shall give priority to users performing exercises.

**Rationale:** Every person prefers to exercise at different times of the day, and therefore the application must be optimized to give priority to users who are currently performing an exercise instead of a user who may be checking his/her activity log since the prior is a much more strenuous task as opposed to the later.

**Fit Criterion:** The system shall cater to a minimum of 1,000 users performing exercises per city at the same time.

**Acceptance Tests:** Test #9

**ID: P7 – Multiple Exercises per Session**

**Description:** The system shall work for multiple exercises within the same session.

**Rationale:** A workout usually consists of multiple exercises done repetitively within intervals, which means that the application should be able to monitor multiple activities within the same session.

**Fit Criterion:** The system shall be able to work with a minimum of 10 exercises per session.

**Acceptance Tests:** Test #10

## Dependability Requirements

### Reliability Requirements

**ID: DP1 – One Fail per Week**

**Description:** The application shall not fail more than once per week per user.

**Rationale:** Applications that are not reliable or crash a lot are not applications that a user would want to return using, but an app that is as strenuous as this must be able to fail as long as it is minimal.

**Fit Criterion:** In this case, the requirement description is quantified, and thus can be tested.

**Acceptance Tests:** Test #11

**ID: DP2 – Fail Safe**

**Description:** The application must fail in safe manner.

**Rationale:** Data is the most important aspect of providing feedback to the user, and therefore a fail-safe must be defined in order to safely store any data that was currently being tracked or used.

**Fit Criterion:** All data prior to a failure must be cached locally to the device before failing before 95% of reported failures.

**Acceptance Tests:** Test #11

**ID: DP3 - User Video Processing**

**Description:** The application must provide feedback on the exercise being performed most of the time.

**Rationale:** The application’s main feature is to provide feedback on the user performing the exercise. If this feature does not work reliably users will not want to use the application.

**Fit Criterion:** The application must provide good reliable feedback on the video over 95% of the time.

**Acceptance Tests:** Test #12

### Availability Requirements

**ID: DP4 – 5 Days of Unavailability per Year**

**Description:** The product exercise functionality shall be available for use almost all the time.

**Rationale:** The application relies heavily on a database to store user data and a machine learning database to provide analysis and feedback of any exercise. Allowing 5 days a year for down time should be enough to handle any failures, database maintenance, and/or machine learning updates.

**Fit Criterion:** The device application should be available 24 hours per day, 360 days a year.

**Acceptance Tests:** Test # 13

### Robustness or Fault-Tolerance Requirements

**ID: DP5 – Available Offline 24/7**

**Description:** The application must be available for use locally all the time.

**Rationale:** The device application has access to data that is also cached locally on the device, and therefore should still be accessible even when the device does not have access to the internet, the central server or the database.

**Fit Criterion:** In this case, the requirement description is absolute, and thus can be tested.

**Acceptance Tests:** Test #13

### Safety-Critical Requirements

**ID: DP6 – Avoid User Injury**

**Description:** The system shall not injure a user through exercise recommendation.

**Rationale:** Too much weight in an exercise can lead to injury. Having a maximum difference of five pounds is heavy enough to produce the desired outcome with a low chance of causing injury.

**Fit Criterion:** There should be a maximum of 10% of reported user injuries.

**Acceptance Tests:** Test #14

**ID: DP7 – Avoid Device Harm**

**Description:** The product shall not overheat the user’s device.

**Rationale:** Using the devices camera, making requests to a database, and also receiving information from a machine learning API can greatly impact the CPU of the device. The application should not harm the user’s device by overheating.

**Fit Criterion:** The application should satisfy the requirements for battery consumption listed through the documentation of supported devices.

**Acceptance Tests:** Test #15

## Maintainability and Supportability Requirements

### Maintenance Requirements

**ID: M1 – Regular Updates**

**Description:** The system must provide updates to the device application, external database, and machine learning API on a regular basis.

**Rationale:** Problems occur when an application is rarely updated, or when it’s only updated during a patch. Having a set schedule for maintenance updates will ensure the product satisfies reliability and availability concerns.

**Fit Criterion:** Regular maintenance updates should be performed once a month for each system.

**Acceptance Tests:** Test #16

**ID: M2 – Well Documented Updates**

**Description:** All maintenance updates should be well documented.

**Rationale:** Having a system that is only able to be updated by a certain number of individuals is detrimental in the long run in the case of losing said individuals or expanding the maintenance team.

**Fit Criterion:** Any developer should be able to understand how the system has changed since the last update without asking for clarification from the author 90% of the time.

**Acceptance Tests:** Test #16

### Supportability Requirements

**ID: M3 – Phone Assistance 24/7**

**Description:** Phone help supportability should be available 24 hours a day, 365 days a year.

**Rationale:** Every person prefers to exercise at different times of the day, and therefore there should always be a real person who is able to provide help to any user who may need it.

**Fit Criterion:** In this case, the requirement description is quantified, and thus can be tested.

**Acceptance Tests:** Test #17

**ID: M4 – FAQ**

**Description:** Online FAQ support should be available all the time.

**Rationale:** Most people usually have very similar questions and thus can make use of a frequently asked questions section to help assist application usage.

**Fit Criterion:** Any question that is asked through phone help support five times per week should be updated to the FAQ section.

**Acceptance Tests:** Test #17

### Adaptability Requirements

**ID: M5 – Device Compatibility**

**Description:** The product is expected to run on Android and iOS devices.

**Rationale:** The client wants as many customers as possible to be able to use the base version of the application without having to purchase additional hardware.

**Fit Criterion:** The software of the product shall be downloadable on a minimum of 85% devices for each respective app store.

**Acceptance Tests:** Test #18

**ID: M6 – Operating System Compatibility**

**Description:** The product is expected to be compatible with any future device OS update.

**Rationale:** Changes to Android OS or iOS sometimes means that applications must be updated to fit the OS’s new requirements in order to be compatible to devices running the new OS.

**Fit Criterion:** The product must be compatible within one month of the OS release.

**Acceptance Tests:** Test #18

### Scalability or Extensibility Requirements

**ID: M7 – Amount of Active Users at a Time**

**Description:** The product shall be capable of processing 1,000 users per city at any given time. This number is expected to increase to 5,000 users per city at any given time within five years.

**Rationale:** The more popular the product becomes, the more users will be using it, which means the active user capacity must eventually be increased in order to handle the volume of active users.

**Fit Criterion:** In this case, the requirement description is quantified, and thus can be tested.

**Acceptance Tests:** Test #9

**ID: M8 – Amount of Exercises per Session**

**Description:** The product shall be able to process twenty exercises per session within three years of its launch.

**Rationale:** Being able to support as many exercises per session is very important to satisfy the population of athletes who train large amounts of time while doing various different exercises.

**Fit Criterion:** In this case, the requirement description is absolute, and thus can be tested.

**Acceptance Tests:** Test #10

### Longevity Requirements

**ID: M9 – Operational Length**

**Description:** The product shall be expected to operate within the maximum maintenance budget for a minimum of five years.

**Rationale:** Previous expectations and estimations of releases has been based upon a five-year basis, therefore having the product live for a minimum of five years is expected. The obesity rate of the United States is also growing, which means the amount of individuals who will need an avenue to be critiqued on his/her exercise will also grow.

**Fit Criterion:** In this case, the requirement description is absolute, and thus can be tested.

**Acceptance Tests:** Test #19

## Security Requirements

### Access Requirements

**ID: S1 - User Login**

**Description:** User data should only be able to be accessed by staff employees.

**Rationale:** User data should be kept as private as possible. The Get Fit application stores medical information of the users, so it is imperative that information is kept confidential.

**Fit Criterion:** Accounts that are not staff accounts should not have the ability to login to accounts other than their own.

**Acceptance Tests:** Test #20

**ID: S2 - Staff Login**

**Description:** Staff accounts can only be accessed by manager accounts

**Rationale:** Most company breaches of security are caused by disgruntled employees. In the case of employment termination, or misuse of account access, managers should be able to access staff accounts to be able to shut them down

**Fit Criterion:** Managers should be able to access the accounts of employees on their staff

**Acceptance Tests:** Test #20

### Integrity Requirements

**ID: S3 - Quality Assurance**

**Description:** The product shall protect against misadvising the users of the Get Fit Right application.

**Rationale:** The Get Fit application gives advice to help users safely lift heavy weights, so it is imperative the information the user is getting is accurate.

**Fit Criterion:** Quality Assurance tests with the Get Fit Right fitness experts, which monitor the types of plans the machine learning algorithm prescribes. The application gives users proper feedback 95% of the time.

**Acceptance Tests:** Test #12

### Privacy Requirements

**ID: S4 - Information Policy**

**Description:** The product shall be transparent as to how it uses user information

**Rationale:** It is important for the application to implement language which allows users to understand how their information is being used in order to comply with the law and avoid any legal penalties.

**Fit Criterion:** The users will have to agree to an accordance policy before utilizing the application as well as having to agree to an accordance policy per each update.

**Acceptance Tests:** Test #21

### Audit Requirements

**ID: S5 - Records**

**Description:** The product shall retain records of plans prescribed to users

**Rationale:** This is necessary in order to audit the application and make sure the machine learning algorithm is giving quality advice

**Fit Criterion:** User information should be stored in a database and accessible for review to the proper authorities when it becomes time to audit the application

**Acceptance Tests:** Test #22

### Immunity Requirements

**ID: S6 - Vulnerabilities**

**Description:** Build a product that is secure against hackers and malicious software

**Rationale:** The Get Fit Right application contains sensitive data on users which must be kept secure

**Fit Criterion:** Penetration tests from a team of security experts will be necessary in order to test the application for vulnerabilities

**Acceptance Tests:** Test #23

## Usability and Humanity Requirements

### Ease of Use Requirements

**ID: A1 - User Age Usability**

**Description:** The product shall be easy for all ages – teenagers to seniors – to use.

**Rationale:** It’s recommended that every person exercise at least thirty minutes a day, and therefore the product should be suitable for all ages.

**Fit Criterion:** Tracking statistics will be used to track various variables, like the number of clicks or amount of times needed to achieve a certain goal per user based on age. The variance should roughly be the same for all ages, which should be less than 10% of users.

**Acceptance Tests:** Test #24

**ID: A2 - User Satisfaction**

**Description:** The product shall make the user want to use it.

**Rationale:** A person who enjoys using an application will use it more, and the aim of this application should be to help as many people as possible.

**Fit Criterion:** Constant monthly requests for satisfaction ratings will be displayed to the user. The average satisfaction level should remain above 85%.

**Acceptance Tests:** Test #25

### Personalization and Internationalization Requirements*.*

**ID: A3 - Measurement System**

**Description:** The product shall adhere to the measurement system the user sets, otherwise it will use the measurement system used by the country he/she is currently in.

**Rationale:** The product should ensure that the user does not have to struggle with the cultural conventions of the developers.

**Fit Criterion:** The number of users that change his/her measurement settings in a month should remain under 5% of monthly active users.

**Acceptance Tests:** Test #26

**ID: A4 - Language Support**

**Description:** The product shall support multiple languages.

**Rationale:** The product should not discriminate against clients who may not fully be comfortable with the language used by the developer, and therefore should be able to choose the language he/she prefers to use.

**Fit Criterion:** the number of users that change his/her language settings in a month should remain under 5% of the monthly active users.

**Acceptance Tests:** Test #27

### Learning Requirements

**ID: A5 - Learning Curve must be Small**

**Description:** The system shall be easy enough so that new users can easily understand how to navigate the application.

**Rationale:** An application that is not easy to use is an application that a user will not return to.

**Fit Criterion:** 90% of users should be able to load an exercise routine using 40% of the time he/she originally took when first using the application.

**Acceptance Tests:** Test #24

### Understandability and Politeness Requirements

**ID: A6 - Simple Language**

**Description:** The product shall use terminology that is understood by most people in the loaded language.

**Rationale:** The application is aimed toward professionals and novices, therefore it should only use word and terms that anyone can understand.

**Fit Criterion:** Only words in a high school dictionary can be utilized within the application.

**Acceptance Tests:** Test #28

**ID: A7 - Understandable Messages**

**Description:** The system shall hide the details of error, loading, and construction occurrences, but still give a rough idea of what’s going on.

**Rationale:** A user does not need to understand the inner workings of the application.

**Fit Criterion:**  Only messages chosen through a blind study test can be used for output during said occurrences.

**Acceptance Tests:** Test #29

### Accessibility Requirements

**ID: A8 - Individuals with Disabilities**

**Description:** The product should be accessible to individuals with disabilities.

**Rationale:** It is self-defeating to exclude a sizeable community of potential customers.

**Fit Criterion:** The product shall conform to the Americans with Disabilities Act.

**Acceptance Tests:** Test #24

**ID: A9 - Zoom for the Visually Impaired**

**Description:** The product shall include a zoom mode for use by visually impaired users or the elderly.

**Rationale:** Some users find it hard to see “normal” font and thus need a large, easier font to be able to read without strain.

**Fit Criterion:** The zoom shall conform to the zoom feature implemented within the device being used.

**Acceptance Tests:** Test #24

### User Documentation Requirements

**ID: A10 - Application Description**

**Description:** The product shall have a short description on the respective app store accurately explaining what the product is.

**Rationale:** A potential user will most likely download the application if he/she is interested in what the application has to offer based on the description.

**Fit Criterion:** A survey of 500 individuals will be conducted to pick the best description out of three options. The description that scores the highest will be chosen.

**Acceptance Tests:** Test #30

**ID: A11 - Installation Guide**

**Description:** The product shall have a well-documented digital installation guide.

**Rationale:** Setting up the application should be as easy as installing it.

**Fit Criterion:** A test of 500 users will be given different installation guides. The guide that produces the fastest average time will be used.

**Acceptance Tests:** Test #31

**ID: A12 - Frequently Asked Questions**

**Description:** Frequently Asked Question section will be dedicated to the website of the product.

**Rationale:** Most people usually have very similar questions and thus can make use of a frequently asked questions section to help assist application usage.

**Fit Criterion:** Any question that is asked through phone help support five times per week should updated to the FAQ section.

**Acceptance Tests:** Test #17

### Training Requirements

**ID: A13 - Online Demonstration Videos**

**Description:** The product shall have online training videos on how to use the application and how to perform the recommended exercises.

**Rationale:** Reading something and trying it is a different experience from watching someone else do something and then trying to repeat it.

**Fit Criterion:** Videos should be released before the product is released, and each video must be updated based upon any changes to the application/outdated videos.

**Acceptance Tests:** Test #2

## Look and Feel Requirements

### Appearance Requirements

**ID: L1 - Aesthetic**

**Description:** The product shall be visually appealing to users of all ages

**Rationale:** The application is not meant to target a specific age demographic therefore it should be appealing to users of any age

**Fit Criterion:** Focus groups and feedback from the mobile phone app markets will be used to gauge the effectiveness of the UI

**Acceptance Tests:** Test #32

**ID: L2 - Simplicity**

**Description:** The product shall develop an interface which displays information in an easy to understand way

**Rationale:** The application is meant to give instruction to users, and therefore should be as simple to understand as possible as to maximize the effectiveness of the application

**Fit Criterion:** Focus groups and feedback from the mobile phone app markets will be used to gauge the effectiveness of the UI

**Acceptance Tests:** Test #32

### Style Requirements

**ID: L3 - Ethos Appeal**

**Description:** The application should appear credible

**Rationale:** The application is meant to give advice and therefore should impose a sense of trust to the users

**Fit Criterion:** Use data analytics to monitor the users who come back to the application for further use versus those who are lost after account creation. User return rate is greater than 60%.

**Acceptance Tests:** Test #33

## Operational and Environmental Requirements

### Expected Physical Environment

**ID: O1 - Location**

**Description:** This product should be used by a user located in a gym or somewhere with access to weights

**Rationale:** The application is meant to correct weightlifting form so having access to the proper equipment is necessary

**Fit Criterion:** The application should not have any problems analyzing a user performing an exercise whilst utilizing the proper equipment.

**Acceptance Tests:** Test #34

### Requirements for Interfacing with Adjacent Systems

**ID: O2 - Mobile Devices**

**Description:** This product shall work for both Android and IOS devices.

**Rationale:** The product should not be limited to those with a certain device as most all smartphones have the technology which make them capable to run this application.

**Fit Criterion:** The application should be release to both Android and IOS markets and will receive updates at most a week apart from each other. Mobile Devices will be the medium on which the application operates.

**Acceptance Tests:** Test #18

### Productization Requirements

**ID: O3 - Storage**

**Description:** The product should take little space on the user’s device.

**Rationale:** The application should be able to be easily installed for users of any device.

**Fit Criterion:** The application should take little space on the user’s device as to be accessible to widest range of audience. Most of the infrastructure will be stored and managed on the cloud, so the user should not have to clear space for the application to be installed. The application takes less than 30MB.

**Acceptance Tests:** Test #18

### Release Requirements

**ID: O4 - Release Cycle Obligations**

**Description:** Updates to the application will be at least once on the first of every month.

**Rationale:** Updates will be necessary as to not only make updates from a performance perspective, but in order to implement security patches on a regular basis.

**Fit Criterion:** The application will notify users of an update available whenever there is one available. Users will be notified and given a few weeks to update the application before they get locked out. The reason for this is to maintain security and support only the most current releases of the application.

**Acceptance Tests:** Test #18

## Cultural and Political Requirements

### Cultural Requirements

**ID: C1 - Kind Towards All**

**Description:** The application shall not be offensive to religious or ethnic groups.

**Rationale:** The users will be from all around the world and it is not okay to discriminate against someone based on race, color, gender, etc.

**Fit Criterion:** Any simulations or videos that are displayed to demonstrate an exercise, the people in the simulation or videos should be modestly dressed and not be wearing any offensive clothing. All text displayed in the application should not have any derogatory connotation to a specific group.

**Acceptance Tests:** Test #28

**ID: C2 - Multiple Languages**

**Description:** The application shall display text and directions in English, Spanish, French, Chinese, and Hindi.

**Rationale:** The users will be from all around the world and it is imperative that users be able to use the application in the language of their choice.

**Fit Criterion:** All content in the app from user information to workout data to simulations and videos should be available in the languages mentioned above. The user should be able pick the language of the choice. English will be the default language.

**Acceptance Tests:** Test #27

**ID: C3 - Standard and Metric Units**

**Description:** The application shall display all numbers and other information in either standard or metric units of measurement.

**Rationale:** The users will be from all around the world and different users use different units of measurements.

**Fit Criterion:** All content in the app that deals with units of measurement would be able to be displayed in either standard or metric units of measurement. The user should be able to pick between the two options.

**Acceptance Tests:** Test #26

### Political Requirements

**ID: C4 - Client Brand Recognition**

**Description:** The client’s branding should be clearly visible in multiple areas of the application.

**Rationale:** The client will be investing a lot of money in developing this application and needs to have proper recognition when the users are using the application.

**Fit Criterion:** The client’s logo should be displayed on the splash screen, on the navigation drawer, and the about us section of the application. The application should also include the tagline at the bottom of the main application screen. The copyright information should also be displayed in the about us section.

**Acceptance Tests:** Test #35

**ID: C5 - Third Party Brand Recognition**

**Description:** Components in the applications that rely on third party software, should be properly recognized.

**Rationale:** Third party software should be used in accordance with any licenses and should be properly accredited in the application.

**Fit Criterion:** The application should display proper logos, taglines, and copyright information for third party software. When YouTube videos are shown to the user to describe and demonstrate and exercises the YouTube logo should be shown. Any articles or videos hosted on other websites that are shown to the user for demonstration or information purposes should also be given proper credit by displaying their logos and other information based on the agreement.

**Acceptance Tests:** Test #35

**ID: C6 - Unique Application Features**

**Description:** Components in the applications should differ from competition applications in functionality and looks.

**Rationale:** The application should be different from pre-existing applications in the market in order for the application to gain popularity and avoid any potential lawsuits from competitors.

**Fit Criterion:** The application should not blatantly copy features from competition applications. Color schemes and other visual features should be distinct and unique. Features that are common with other applications should visually be different and make the functionality smooth as possible.

**Acceptance Tests:** Test #36

## Legal Requirements

### Compliance Requirements

**ID: G1 - Secure Personal Data**

**Description:** The application should safely and securely store personal user data and comply with the Data Protection Act.

**Rationale:** The application should be secure enough, so the users are comfortable sharing personal information and allowing the application to store their workout data.

**Fit Criterion:** All personal data should be encrypted and stored on secure servers. User data should not be sold or shared with any third-party members. A lawyer should check that application complies with the Data Protection Act.

**Acceptance Tests:** Test #21

**ID: G2 - Third Party Licenses**

**Description:** The application should only use third party applications that are properly licensed.

**Rationale:** The application should use only licensed software in order to avoid lawsuits from those companies.

**Fit Criterion:** Licenses should be acquired before the application is developed. A legal team should verify the licenses and ensure they are renewed in a timely fashion.

**Acceptance Tests:** Test #36

### Standards Requirements

**ID: G3 - Nike Inc. Standard**

**Description:** The application should be a good representation of the Nike Inc. brand.

**Rationale:** The application will be marketed as a product of the Nike Inc. and should have the look and feel of such a highly acclaimed company.

**Fit Criterion:** The application should match the reliability and consistency of the Nike Inc. brand and remind the users of the brand they love.

**Acceptance Tests:** Test #33 and Test #35

**ID: G4 - Industry Standards Review**

**Description:** The application should comply with iOS and Android application standards.

**Rationale:** The application should be reviewed so that it can be successfully deployed to the App Store and the Play Store.

**Fit Criterion:** A team should review the codebases from each native application to ensure that they comply with the industry standard for native applications, as well as the standards of the App Store and the Play Store.

**Acceptance Tests:** Test #18

## Requirements Acceptance Tests

### Requirements – Test Correspondence Summary

Table is located in the G12CS440GetFitRightCorrespondenceSummary.xlsx file.

### Acceptance Test Descriptions*.*

**ID #1 – Sign In**

**Description:** Registered users can login. Invalid credentials are prompted to make an account.

**ID #2 – Enter Workout**

**Description:** User is shown the current workout. Current workout is analyzed by processing the user video and the user is given feedback. The user is shown demonstration videos for the specific workout.

**ID #3 – Get Workout**

**Description:** Right workout data is loaded and displayed to the user.

**ID #4 – Application Sync**

**Description:** Application updates the workout time elapsed and gets data from the machine learning database once every second.

**ID #5 – Bootup**

**Description:** The application starts up in 2 seconds 85% of the time.

**ID #6 – Live video processing**

**Description:** The video of the user working out is processed in less than five seconds on average. A progress bar is shown for the processing.

**ID #7 – Weight Recommendations**

**Description:** The user is satisfied with their weight recommendation over 90% of the time.

**ID #8 – User Workout Feedback**

**Description:** The user reports that they are satisfied with the level of difficulty of their workout over 95% of the time.

**ID #9 – Current active useer**

**Description:** The application caters to over 1,000 people per city at the same time and will support over 5,000 people per city at the same in five years.

**ID #10 – Number of Exercises**

**Description:** The application handles up to or greater than 10 exercises per session and will handle 20 or more exercises per session in three years.

**ID #11 – Application Crashes**

**Description:** The application crashes at most once per week per user and data is properly retained for more than 95% of those crashes.

**ID #12 – Exercise Feedback**

**Description:** Application provides relevant feedback and avoids misadvising from user inputted videos over 95% of the time.

**ID #13 – Application Availability**

**Description:** Full application features are available for 24 hours a day 360 days a year. Offline features are available 24 hours a day 365 days a year.

**ID #14 – User Injury**

**Description:** The application should have less than 10% of reported injuries from exercise recommendation.

**ID #15 – Device Harm**

**Description:** The application does not overheat the device its running on.

**ID #16 – Application Updates**

**Description:** The application is updated at least once a month, and updates are well documented so other developers don’t need clarification from authors over 90% of the time.

**ID #17 – Application Support**

**Description:** Phone support and in-application FAQ are available at all times.

**ID #18 – Application OS**

**Description:** The application works with 85% of current devices on iOS and Android and will work with all future versions of both OS’s. Updates on both platforms are within a week of each other. The application takes less than 30MB on the users device. The application complies with both the App Store and Play Store standards.

**ID #19 – Budget**

**Description:** The project stays in budget for a minimum of five years.

**ID #20 – Account**

**Description:** User can only access their own account. Staff can access their account and user accounts. Managers can access their account, and staff and user accounts.

**ID #21 – Data Privacy and Transparency**

**Description:** The users are shown an accordance policy to accept when signing up. User data is handled safely and in accordance to the law as verified by a lawyer. The user is notified how the data will be used.

**ID #22 – Records**

**Description:** The application has information regarding all the subscriptions of the users.

**ID #23 – Vulnerabilities and Data Threats**

**Description:** The application is safe from all types of hacking and breaching.

**ID #24 – Age Usability**

**Description:** The variance for the number of clicks to do a certain task by all age groups is less than 10%. 90% of users can start an exercise routing in less than 40% of the time than they initially took. The application complies with the Americans with Disability Act and allows users who are visually impaired to zoom in and view content bigger.

**ID #25 – User Satisfaction**

**Description:** The application satisfaction level by in application surveys is above 85%.

**ID #26 – Units of measurement**

**Description:** The application sets the units of measurement based on location. Less than 5% of the users need to change their preferences. The user is given the option to change their setting.

**ID #27 – Language Support**

**Description:** The application set the language based on user location. Less than 5% of users need to change their language settings. The user can pick between any language by downloading language packs in the application.

**ID #28 – Simple and Proper Language and Instruction.**

**Description:** The application uses only high school level dictionary words. The words and instruction are not offensive to any cultural or religious groups or towards a specific individual. Videos shown do not discriminate or have discriminatory parts.

**ID #29 – Error Messages**

**Description:** The application hides error messages, unless pertaining to the user.

**ID #30 – Description**

**Description:** The application uses a description which is deemed to be the best from a survey of 500 random individuals.

**ID #31 – Installation Guide**

**Description:** The application has a installation guide which is deemed to be the best from a survey of 500 random individuals.

**ID #32 – Visual Style**

**Description:** The application is visually appealing and simple based on a feedback from focus groups and survey data.

**ID #33 – Credible and Professional Feel**

**Description:** The application tracks user return rate to ensure the return rate is over 60%. The application gives the feel that it is a Nike Inc. application.

**ID #34 – Application Location**

**Description:** The application can be run on devices in gyms and while the user is performing exercises.

**ID #35 – Brand Recognition**

**Description:** The application displays the Nike Logo, Tagline, and copyright info as needed. The application also displayed 3rd party logo, copyright information based on the agreement.

**ID #36 – Lawsuit Prevention**

**Description:** The application doesn’t copy from competition applications. Visually different from other application. All 3rd party software has proper licensing verified by the legal team.

# Design

## Design Goals

Here are the primary design goals:

Speed and accuracy should be priority over security

The application needs to be able to process the user’s video in real time to give instantaneous feedback. The user is using this to enhance their workout and if the feedback isn’t quick enough, this will deter the user’s workout more that it will help. If the user get frustrated they might stop using the application, which would be very detrimental to this application as a whole. Security isn’t much of a concern with this application as we aren’t dealing with information that is critically sensitive. However, personal user data should be stored in secure databases and other users shouldn’t be able to access other user’s data.

UI is simple, follows a color scheme, and easy to use.

The application should have all features visible to the user readily. For all features that aren’t accessible from the home screen of the application, there should be a tutorial that walks the user through those features of the app. All data and feedback should be shown in a simple layout which avoid overcrowding with not so important information. The user should be able to customize their experience by selecting what information they would like to see. The colors in the application should be subtle to the eye, and should not use contrasting colors. The application should also be friendly to color blind users, and should also tend to users with other disabilities.

Fast reconnection to server in event of connection loss, or local mode.

This is in relation to the first part with speed over security. There are times where certain gyms or other exercise locations might not have the best network connection. When there is poor network connection the application should be able to function in an offline mode. Most features of the app should still be available besides the posture recognition as that relies heavily on an external database and it would be impossible to store all that information on the user’s device. In situations with poor data connection, the analysis should be done in the background so the user can use other features of the app or continue on with another exercise in the time being.

All relevant user data should be saved properly.

The user’s workout data and other personal data should be stored securely and safely in our databases. All this information should be available to the user whenever the user requests for it. The user should have complete control if at any point they would like to delete any information from our databases. The application should only store relevant data, and doesn’t need to save user’s videos that are sent to the applications servers for analysis.

Feedback on exercise should be accurate as possible.

For all exercises the feedback given should be approved by a team of physicians who are experts in the field. All posture data should be verified to be correct before it used in analysis. Suggestions from the application will be followed seriously by many users and this feedback should limit injury and other negatives of exercise rather than making the issues more severe.

The application should be available to the most users possible.

The application should be available on all major platforms and cover most versions. Additionally, all types of users should be able to understand the information being presented. This implies the feedback presented should be in layman terms and should also include the higher level more scientific wording.

## Current System Design

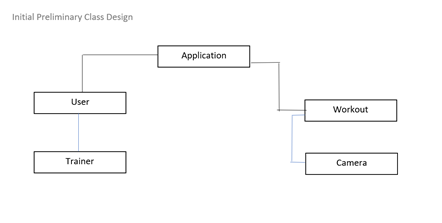
No pre-existing system.

## Proposed System Design

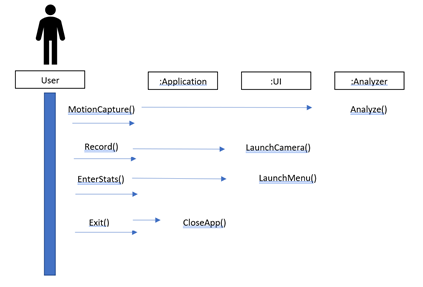
### Initial System Analysis and Class Identification

Most needed Classes:

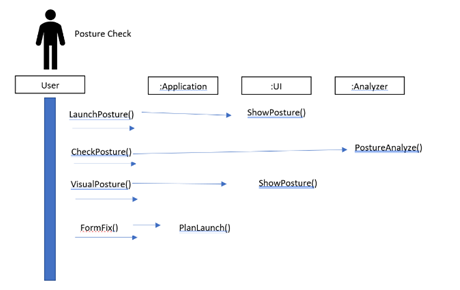
* User (Noun)
  + Representing the user of the application
* Trainer (Noun)
  + Representing the training experts of the application
* Camera (Noun)
  + Representing the main hardware asset of the application
* Workout (Noun)
  + Representing the main objective to be completed by the application
* Application (Noun)
  + The overarching enterprise which encapsulates all other classes

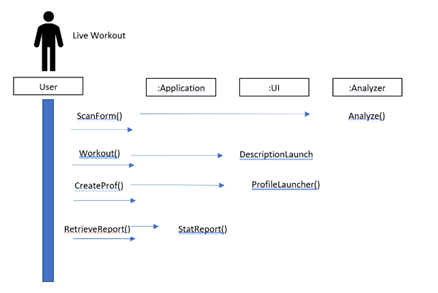
Initial Class Design:

### Dynamic Modelling of Use-Cases



Type of workout





### Proposed System Architecture

The Get Fit Application Utilizes a Model View Controller architectural pattern. The Model component of the application retrieves the user statistics and sends it to the database. The View component of the application is responsible for the user interface section of the application and will handle all the forms in and menus. The Controller component will process information between the last two components of the application. The Get Fit Application will be utilizing the PostgreSQL Relational Database Management System to store data. This is used by popular companies such as IMDB and Amazon. PostgreSQL is different than MySQL and is better for this application because the Get Fit Application relies heavily on speed. The PostgreSQL relational database management system is known for fast read write speeds which will be crucial for our application.

### https://lh6.googleusercontent.com/cm7ZN9CsRpoqLyNy5-vKdgf_my0pqol-vrq2u8DkG9SguHy7GgnTyVlxgUYkNK15IxhxosTfxNYUlSQhs_GhrQxB6gomcTnGNzRxDa-NFaQUubh-oXqCCIdy11PKpucQsoN2oOOGez0Initial Subsystem Decomposition

There are 4 major subsystems in this project:

Exercises

The exercises subsystem includes the whole tree of workouts, which begin from the Workout class. The workout tree is implemented using the composite design pattern to create an efficient way of adding in other exercises as necessary. Using this design pattern also allows to reuse a lot of code and keeps the whole subsystem organized.

Analyzer

This subsystem includes the Camera, Analyzer, and Accuracy classes. These classes are used to process the user videos and give the user feedback. These classes work with the data in the Workout tree and then give the feedback to the Application class.

User Interface

This subsystem includes the GUI and the Notification classes. These classes are used to display all the data and feedback to the user. This also handles the user’s clicks by passing the event information to the Application class.

User

This subsystem includes the User tree. It is responsible for keeping track of all the information regarding the different users of the application when the application is running. Works with the Application class to enable all the other features of the application.

## Additional Design Considerations

### Hardware / Software Mapping

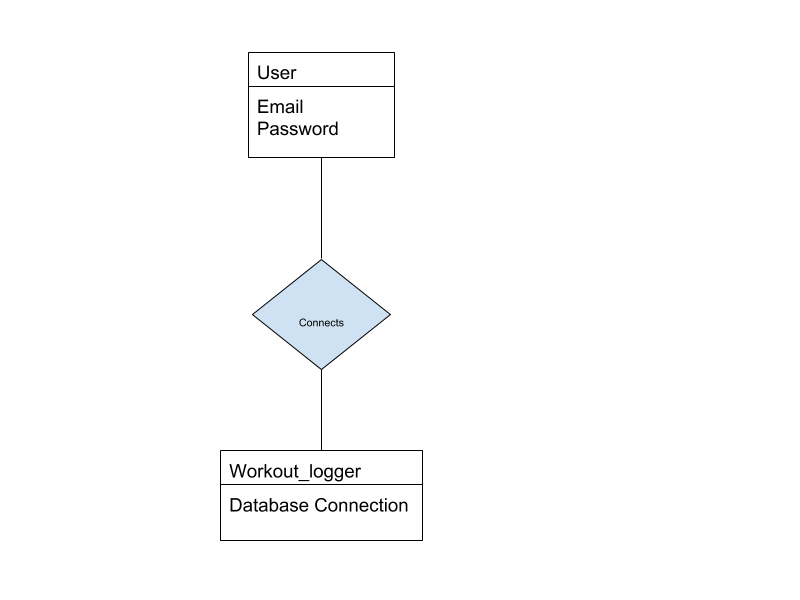
Workout Database

Workout History Database

User Phone

The application connects the user’s device to the databases. The two databases are the workout logger, that stores the user’s previous workouts, and the workout type which store the selected workout being performed and posture data.

### Persistent Data Management



For our product, the only persistent data that we wish to store is the users email and password so that they don’t have to sign into the application when they close the application. This persistent data is stored locally on the user’s device. All other data is stored on the databases. To access all other information the users will need their email and password to obtain that data.

### Access Control and Security

The security subsystem implementation will be handled utilizing the cloud partner platforms tools that will be on hand and as such there will not be a dedicated product subsystem for security. Instead there will be classes to handle how connections are made to databases from the user’s device to our cloud partners. No new classes are needed as the secure connections, as all databases calls will be made securely with pre-existing classes.

### Global Software Control

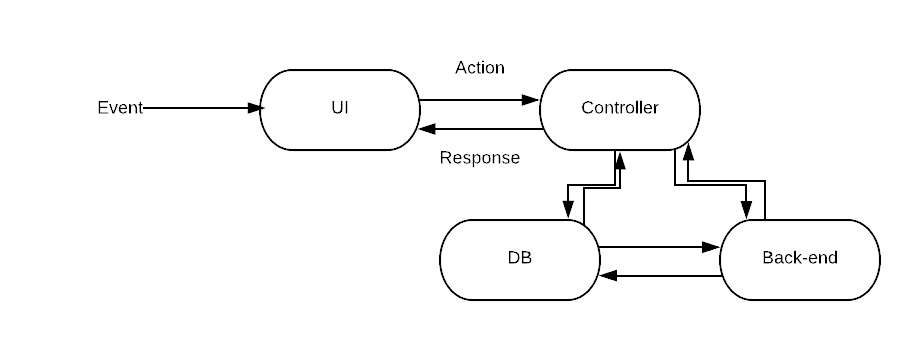
Due to our product being a phone application we must use multiple threads to handle communication between the user’s device and our databases. Threads will also be used to handle computational tasks like comparing the users posture data to that of the correct workout posture data as to not freeze the user’s device UI and cause a negative user experience. A new class that needs to be added to handle this is the controller class. This controller class will handle all the multi-threading and will then show the relevant results when the threads complete that specific computation.

### Boundary Conditions

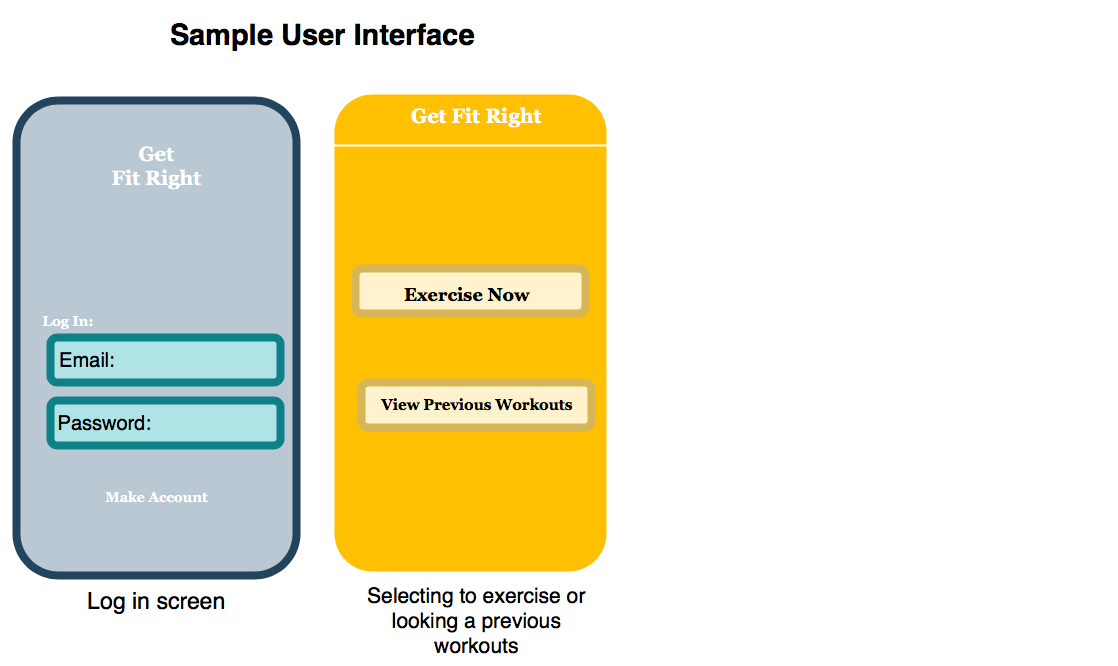
As to obtaining the users information during account creation we plan to offer text fields and buttons to enter that information. These objects will also be offered when logging in if the user already has an account and the entered data will be validated against data from the database to verify if the information is correct and if the user is to be signed in or not. Notification system will also be used to display analysis on workout being performed. No new classes are needed to handle the boundary conditions as they will be handled by the controller, application, and GUI classes.

### User Interface

User interface has to be well designed and structured as to be easy to use while working out and looking at previous workout information. It must also be nice looking to be able to attract new users and keep existing users engaged in the product. First time opening up the application users will be prompted to either log into the application or create an account to use the application. Once logged in users will have to options choose a workout and then perform it while our product analysis it in real time to see if it being performed, or they can choose to look at their previous workout data to see improvements in the number of sets, reps, weight, or correctness of their workouts.



In the above example we have a subsystem diagram where the user selects to performs an action, in the UI, like selecting a workout to perform. Then we communicate using a controller subsystem, that handles our threading as to not freeze the UI, thus connecting information we receive from the database and sending it to the backend where we analyze the information and send the response back to the UI so that the user can see in real time if they are performing the workout correctly.



### Application of Design Patterns

Not applicable as all patterns that will be used have already been covered in previous sections.

## Final System Design

In our final system design we added the controller class that houses our code that handles the threading in our applications. This helps facilitate the systems operations by having a controller that handles what the task the system should be performing when a user selects an option in their device. Also it connects the workout logger class that will house non-detailed local variations of the data requested by the user as to limit the number of read and writes to our databases and speed the process of looking at older workouts by not having to send the request and wait for the response every time the user initiates the process of looking at the older workouts. If the user selects one of the previous workouts that is displayed, then a request is sent to our database to obtain more detailed data for that particular workout to be displayed to the user. As mentioned, before it helps keep traffic to our databases lower and also helps the users have a more pleasant experience by displaying the information even when the user has bad network connection and limiting the amount of data required to use the application.

## Object Design

### Packages

Presentation: All aspects of the product that the user views, interacts with, or needs in order to navigate the system.

Data Access: All aspects of the infrastructure of the product that is needed in order to allow the system to work as intended.

High Computation: All aspects of the product that rely on heavy persistent data, machine learning, artificial intelligence, and cloud services in order to achieve the goal of the system.

### Subsystem I

Presentation Layer

* Front End: application, runnable, start, view, change screen, interact, select
* Data Interaction: start, select, interact, update, pull, push, refresh, sync,
* Camera Interaction: start, select, permission, update, record, capture, refresh, sync

### Subsystem II

Data Access Layer

* Data Storage: save, update, sync, archive, refresh, security, lock, disconnect
* Data Retrieval: load, update, sync, connect, reconnect, save, unlock
* Camera Manipulation: record, analyze, mark, upload, sync, pause

### Subsystem III

High Computation Layer

* Database: create, read, update, delete
* Application User Interface: index, new, create, show, edit, update, destroy
* Analysis System: feature detection/selection/extraction, adapt, preprocess, process, postprocess, encode, regression, classification
* Cloud: AWS, Apache, Content Delivery Network, Storage, Sourcing, Backup, Restore

# Project Issues

## Open Issues

Here are some of the following the open issues regarding the product:

1. Posture data required for analysis hasn’t been reviewed by medical professionals.
   1. The data required to process user’s videos need to be properly reviewed to ensure that the proper recommendations are given to the users. A team of physicians, doctors, and sports trainers will need to review data for different exercises.
2. The feasibility of the application running on older devices hasn’t been considered.
   1. The major feature of this application is the posture analysis for exercises. This requires a good quality video and processor to handle the application. This may not be possible on older devices with average cameras. Further testing needs to be conducted.
3. Possible clothing and hardware to accompany the application is being considered.
   1. For certain exercises a simple video of the exercise might not be enough to give proper feedback and recommendations. In order to tackle this issues special clothing with attached sensor, or other special hardware might need to be developed for the user to get the optimal benefit. Such clothing and hardware is being tested in lab environments to see if the they do actually provide added benefit.

## Off-the-Shelf Solutions

### Ready-Made Products

The application needs to be vertically integrated with the rest of the fitness apps. Despite posture tracking being the main feature, this application should also have basic workout tracking. Rather than building an in house solution for this, features from apps like *Google Fit*, *MyFitnessPal*, and *Runtastic* should be used. These applications have core functionality like workout tracking, whether it be live or user entered, calories and macros tracking, suggestions on how to stay generally fit on a regular basis, and connecting to hardware such as a Fitbit. These features can be used from these apps or the general concept can be recreated by the software developers of this application.

### Reusable Components

Some toolkits that this application can use are the camera API’s from both Google and Apple. The app can also use the Augmented reality API’s as a means to illustrate the exercise to users. Both IOS and Android have pre-built API’s for most basic functionality which could be combined together to create a full-fledged application.

### Products That Can Be Copied

Nike Inc. has an application called *Nike Training Club* which has pre built workouts the user can choose from for a specific day and type of workout. The features of that application can be integrated into this application. The applications mentioned in 29a can also be used with valid licensing, or this new application can be created in partnership with one of those companies.

## New Problems

### Effects on the Current Environment

This application might cause longer wait times for equipment at the gym. Since users will use the application’s feedback, they might take longer times on specific exercises. This may cause a disruption in the flow of a gym. Due to this the product should not take too long to give feedback as this delay feedback might have a wider effect than the individual using the application itself. A study can be performed in a live gym environment to make sure effects of this application are not adverse.

Another issue this might cause is user’s not requiring the use of personal trainers. Since this application could potentially provide the same feedback and workouts as a personal trainer would this might affect the jobs of those individuals.

### Effects on the Installed Systems

Since this application is quite resource heavy with its video recording and data sending it might make an impact on the phone’s battery very heavily. It might also cause the phone to overheat with excess usage times of the camera. In order to avoid this the application should be as efficient as possible to protect the user’s devices.

### Potential User Problems

This application could potentially harm the user if incorrect feedback is given. This is why it is absolutely imperative that the user get proper feedback. The user might also be distracted by the application while working out which may cause them to do the exercise incorrectly and cause injury that way.

### Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

At times certain gyms and exercise environments might not have a proper network connection. This limits the features of the app as our main feature of posture tracking and analyzing requires the use of a data connection. Posture analysis is too intensive and data heavy to be done on a user’s phone so when network connection is poor or not there full features might not be available. Additionally, the servers might also be slowed down when too many request come in. When the number of users grow, it might become difficult to process all videos in real time and more servers might need to be bought to fix that issue.

### Follow-Up Problems

There aren’t any follow up problems at the moment.

## Migration to the New Product

Not Applicable*.*

## Risks

* + Inefficient code in the front end or back end can severely cause the project to underperform as it will slow the product down
  + Having bugs in the code can cause the project to fail due to an excessive amount of application crashes, or even having broken/unexpected functionality.
  + Not allocating resources properly to each layer of production can cause the product to underperform due to lack of funding or unprepared resources.
  + Having security flaws can greatly underperform the product or even cause it to fail as users will not want to use a product that is vulnerable to attacks.
  + Underwhelming data storage/management can cause the product to underperform if the data needed is down or even missing for large amounts of times due to maintenance.
  + Having a user interface that is difficult to navigate or is unpleasing to the users’ eyes can drastically have the product underperform as more users will tend to user other services if they find the product to be difficult to use or unappealing.

## Costs

$5,000,000 is the estimated cost to fully develop the iOS and Android applications, the database system, engineering the architecture, modeling the machine learning algorithms, testing, and advertisements over the course of 3 years. We estimate the number of hours needed to complete the project to take around 4,000 hours per year, give or take 100 hours. Resources include, but not limited to, developer salaries, consulting, technology like computers, technology like servers, rent for a working space, insurance, testing, and advertisement. The team will consist of 3 engineers, one business analyst, and one marketing representative. The total salary for those employees is $350,000. Servers and other hardware required for testing will costs $1,000,000. Consulting reports for calculating feasibility will require $250,000. The rest of the cost will be used in advertising and promotions.

## Waiting Room

* + The ability to use the platform as a social media network to compete with friends
  + The ability to have a one-one-one session with a trainer through the app
  + The ability to have the application sync with smart devices like Apple Watches to more accurately track metrics.
  + The ability to the easily send and share personal data to and from trainers, doctors, family, etc.
  + The ability to track food and water consumption to track caloric intake
  + The ability to have the application recommend meal plans to follow
  + The ability to have the application curate a personal workout plan

## Ideas for Solutions

Some ideas towards the implementation of the databases that the product uses would be to use cloud partner like AWS or Azure to provide quick access, limit downtime, and provide scalability towards our product. The programing language used to implement the product that would be Java, due to it being object-oriented design along with the multithreading aspect that it provides for the product implementation. As such we would want to design our code in such a way that we have high cohesion and low coupling so that code maintenance is easier to maintain, and new features are easier to integrate. As to IDE used for product creation on the android side we plan to use Android Studio and for iOS we plan to use XCode. In order to connect to external devices, existing API’s can be used to have a seamless integration.

## Project Retrospective

Project overall went really well due to the communication between team members and just staying ahead of due dates and not doing everything the day that its due. Also, the fact that all team members agreed on the project topic and were genuinely interested in making a product that would help people workout in a safer manner. As we have not yet developed this product, we did not have a negative experience in which something didn’t work.

# Glossary

“Textbook” – When something is executed perfectly it can be described using this word.

PR – Personal records. Common uses is when an individual reaches a new weight on a lift or a runner completes a race beating their own personal record.

This document did not require much of a glossary as most of the words used were everyday words. For some words definitions were included in the text itself.

# References / Bibliography

|  |  |
| --- | --- |
| [1] | M. Fowler, UML Distilled, Third Edition, Boston: Pearson Education, 2004. |

# Index

Design 51

Requirements 28