

Abstract

The aim of this project is to create a workout logging application, that's easy to use, minimalistic and efficient at recording large amounts of workout data. This application will have a specific focus on weightlifting data, and it should also be easy to use in a gym environment. The users will be able to create and share with others their workout routines, and be able to search for existing workout routines to use as a template that can then be modified by the users if desired. This workout logging application will be accompanied by a website, to enable easy viewing and user analysis of logged workout data, this data will also be sharable. A database will also be used to store the data along with an API to post and get data from the database.

Research into existing products and the user reviews of these products was carried out and then a questionnaire was given to potential users of this product. This information was then used to create a set of requirements that helped to inform decisions on the tools to use, the design of and ultimately the implementation of the application and the website.

Contents

Abstract.....	1
1.0 Introduction.....	4
1.1 Background.....	4
1.2 Objectives	4
2.0 Information Review	5
2.1 Similar Existing solutions.....	5
2.1.1 FitnessBuilder.....	5
2.1.2 Nsuns 5/3/1 Program Log	6
2.1.3 HeavySet.....	7
2.1.4 Strong: Exercise Gym Log, 5x5.....	8
2.1.5 Summary of Similar applications.....	9
2.2 Approaches to engaging potential users.....	10
2.3 Information review conclusions.....	12
3.0 Requirements	13
3.1 Non-Functional Requirements	13
3.2 Requirements from Information Review	13
3.3 Requirements gathered from questionnaire	14
3.4 Final Requirements	14
4.0 Development tools and approaches.....	15
4.1 Available Mobile Platforms	15
4.1.1 Android	15
4.1.2 iOS	15
4.1.3 Mobile web apps	16
4.1.4 Conclusion	16
4.2 Mobile Development Tools	16
4.2.1 Android Studio.....	16
4.2.2 Apache Cordova.....	17
4.2.3 Xamarin.....	17
4.2.4 Conclusion	17
4.3 Website	17
4.4 APIs	18
4.5 Software development approaches.....	18
4.6 Version Control.....	18
4.7 Conclusion	18
5.0 Design	19

5.1 Mobile app design.....	19
5.2 Database Design.....	21
5.3 Website Design	22
5.4 API Design.....	23
6.0 Implementation	24
6.1 Database.....	24
6.2 API.....	26
6.3 Mobile Application	27
6.4 Website	30
6.6 Local Testing	32
8.0 Evaluation	35
8.1 User Evaluation.....	35
8.1.1 User Feedback.....	35
8.1.2 Summary	36
8.2 Reflection of Requirements	37
8.3 Reflection of aims	39
8.4 Summary.....	39
8.5 Future Work.....	40
9.0 Critical Review	41
References.....	42
Bibliography	46
Appendices.....	51
Appendix A - Questionnaire	51
Appendix B – Questionnaire Results.....	55
Appendix C – User Evaluation Interview Guide	61
Appendix D - User Evaluation Interview Answers.....	61
Appendix E – Project Specification.....	64
Appendix F – Ethics and Risks Checklist.....	66
Appendix G – Stored Procedures with Testing.....	67

1.0 Introduction

1.1 Background

Fitness logging has been around for a long time, but it's evolved in the last 30 years from something done on paper, to something that can be recorded on a computer, and now even to something that can be shared with friends and even automated. With the switch to digital logging it's also allowed for better data analysis and for more community interaction where users can help each other in recommending and modifying workout routines based on goals.

This project will be focusing specifically on weightlifting applications. From initial investigations there is a lot of existing fitness applications that focus on weightlifting but a free to use solution that embodies a simplistic, easy to use interface and that allows you to record and analyse a lot of data is difficult to find. This shows that there could be a gap in the market for a fitness application like this.

1.2 Objectives

With the background chapter in mind the aim of this project is to provide an easy to use, minimalistic Fitness logging application, with a specific focus on weightlifting, which should be capable of recording a range of data supplied by a user. Along with this a website will be built which will be able to show this data in a user friendly intuitive manner, a website is appropriate for this as the aim of the project is to keep the application as minimalistic as possible. This data should allow for users to be able to review their performance in a clear and concise way, if they are not progressing with their current workout routine they should be able to share their data and workout routine to allow for others to offer advice if they wish. Users should also be able to search for someone else's workout routine to use instead of their own, and have the ability to modify the routine to their own needs. Where possible the application should allow for user customisation to allow for a personal experience. To achieve this, a database will be required to store the data and an API we be appropriate to access the data in a consistent manner from both the mobile application and website.

2.0 Information Review

To help inform a user questionnaire, research into some existing fitness applications will be carried out for initial research, and then different types of user questioning will be reviewed and summarised.

2.1 Similar Existing solutions

What is to follow is a detailed look into four existing fitness applications, then this data will be summarised into a table; this will enable the positives and negatives of each application to be easily viewed.

2.1.1 FitnessBuilder

FitnessBuilder (PumpOne, 2017) is a workout logging application with a focus on supporting trainer client relationships. Because of this focus, the application has the ability to share workouts and logged information with personal trainers, an example of this sharing can be seen in figure 2.1. However this information and workout data cannot be shared with friends, this means the application doesn't really have the potential to build a community around it, unlike many other fitness logging applications. There is a wide range of available metrics you can measure using this application, but not much flexibility in being able to swap exercises during your workout.

Customer reviews

This application gets a 3.8/5 rating on Google play, which isn't a bad user score but there is a significant number of 1* reviews. The following is a few examples of these 1* review points that were shared by many users, such as syncing issues "Image sync is terrible" (SH, 2014) in addition to complaints about having to use the pro version of the application "Worked fine but started to tell me I need pro version to see my workouts." (Depwe, 2014). Also complaints about limited exercise selection "I can't believe there are no workouts for the traps" (Snow, 2014). However on the other side a user that gave this application 5* is quoted as saying "You can build workouts on your own or customize the pre-loaded routines" (Anonymous, 2015), this is clearly an important feature to this user and many similar reviews can be found stating the same in different words showing that this is a feature that should be strongly considered when creating similar applications.



Figure 2.1: Application image to show trainer client sharing data source (PumpOne, 2017)

2.1.2 Nsuns 5/3/1 Program Log

Unlike FitnessBuilder (section 2.1.1) Nsuns 5/3/1 (MonstRApps, 2017) is a very basic application with seemingly no-intent to be monetized; instead it is made to support one specific type of workout routine. This enables the application to be very simple to use as you will not be overwhelmed with features. The simplicity of the application is amplified by the nature of the layout of the application, as can be seen in figure 2.2 and 2.3 all the information you could need is

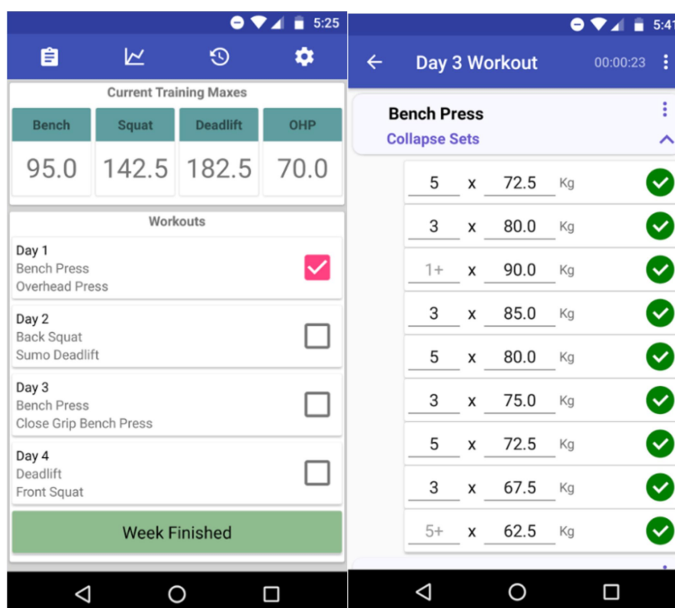


Figure 2.2 & 2.3: Application images of Nsuns to show the layout (MonstRApps, 2017)

shown in a clear and simplistic manner, this combined with the big buttons enables for easy reading and easy data entry when you're at the gym. However this very simplistic application is missing some desirable features it doesn't allow for logging personal data and as it's designed for one particular workout routine, there is also very little flexibility in creating a workout routine. In addition to this there is also no way to back up data from the application online.

Customer reviews

This application seems to be highly regarded by its user base, this seems to be because the application has some simple goals and managed to achieve them well. It's not stretched too thin, it just does what it set out to do and does it well, and this is shown by the 4.8/5 rating it receives on Google Play. A lot of the positive reviews praise the application in a general way for example "Great app. Very easy to use. I love the timer sound and how you can customize it." (B, 2018). However even the generally positive reviews contain some negative points such as "I love this app! Major problems: - you cant add custom exercises - you cant program accessories (fix sets and weights, ask at the end of the week for manual progression) Thanks!" (Pérez, 2017). There are also user complaints about not being able to back up the data, and even losses of data which could have perhaps been prevented with an online backup.

2.1.3 HeavySet

HeavySet (Runloop Ltd, 2018) is an iOS only application specifically designed for tracking weightlifting data. In figure 2.4 you can see what at first glance looks like quite a cluttered screen for inputting data but in actual fact the buttons are big making data entry easy and the fields use auto-predict to guess what you'll type before you have even typed anything, giving the whole experience a smooth feeling. You can also view weights as a

percentage of your one rep maximum lift, giving some extra flexibility. Along with this is a graph right in front of you of your progress on this particular exercise, many more graphs are also available to allow you to analyse your progress. Not un-similar to FitnessBuilder (section 2.1.1) this application allows you to share data, this is shown in figure 2.5, however this is not in app data sharing, you're just given the data as text making the experience not as nice as perhaps an all in app data sharing experience could be, but you can then share this data with whoever you please. Another downfall with this application is a very limited selection of exercises to choose between, which offers the user a reduced number of options when creating their own workouts.

Customer reviews

The users of this application are generally very pleased with it, it receives 4.7/5 rating on iTunes with many users praising it for its UI "The UI is intuitive and organizing workouts is a breeze." (sbthr33, 2018). There are also some negative points though, just like FitnessBuilder (Section 2.1.1) you have to pay for it to be used as advertised "For simply keeping a log of what you lift it is great. The trial period is not quite enough to find that out though." (Timmmm, 2018).

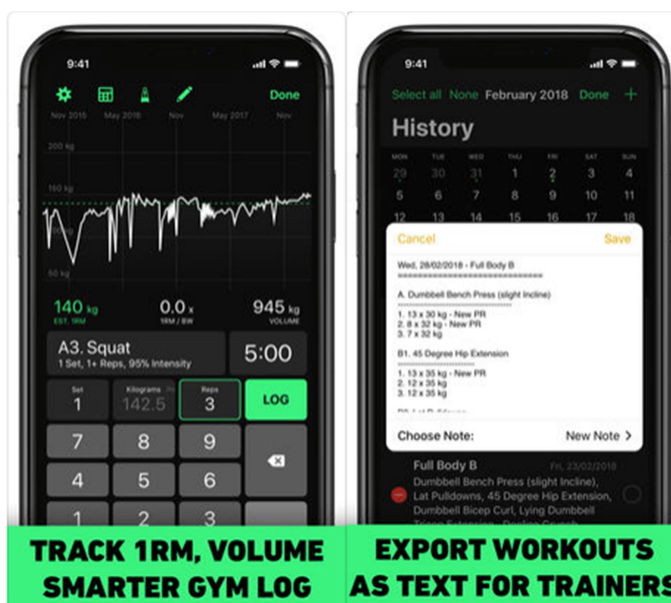


Figure 2.4: Shows the inputting of data (Runloop Ltd, 2018)

Figure 2.5: Shows exporting of data (Runloop Ltd, 2018)

2.1.4 Strong: Exercise Gym Log, 5x5

Strong (Strong Fitness, 2018) is a workout and exercise tracker designed around the workout method called 5x5 (where 5 sets and 5 reps of a given exercise are performed every day) but can be adjustable to many other types of workout methods too. This application has a large library of exercises and as shown in figure 2.6 also contains detailed instructions and shows the target muscle graphically as well as a start and end position of the exercise. In figure

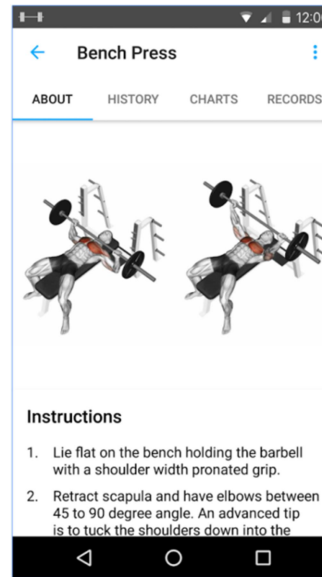


Figure 2.6: Shows exercise instructions (Strong Fitness, 2018)

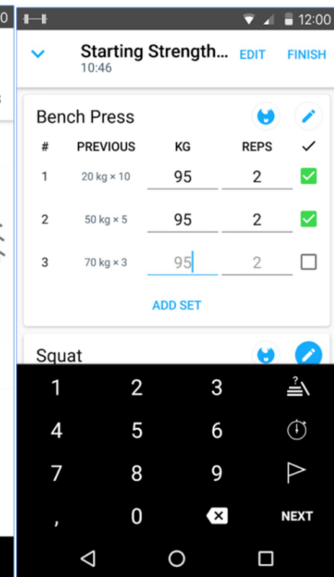


Figure 2.7: Shows the inputting of data (Strong Fitness, 2018)

2.7 the simplistic data entry of the application can be seen where it also shows you as a hint what weight and reps were performed last time you did this exercise. This application also contains a warm up calculator which is unique out of the four applications reviewed here. The data entry looks very similar to the data entry used in Nsuns 5/3/1 (section 2.1.2), which was highly praised by the users of the application. This application like HeavySet (section 2.1.3) also contains many useful graphs, but unlike both HeavySet (section 2.1.3) and FitnessBuilder (section 2.1.1) you are unable to share your data with others.

Customer reviews

This application incorporates so many successful features of the previously discussed fitness applications that it's no surprise it has a 4.8/5 rating on Google Play and a large customer base. The ideas behind how this application has been successful could help my proposed application to be a success. A lot of the user reviews can be represented by these two short reviews "Simple. Effective tool." (Saunders, 2018) "Great tracker, lots of routines and exercises." (lokkheed, 2018). This doesn't mean improvements can't be made however, there are a few complaints about the navigation in the application not being as appropriate as it could be for a gym environment "Could there be an upgrade that prevents the sets or reps you have saved from being accidentally swiped away when try g to enter weight and reps during a workout? I understand the need to exit a routine but when you are in an active exercise session and one is in muscle failure sometimes the set ur trying to enter info on gets deleted with a slight swiping motion" (Waddell, 2018). In addition to this there is also some issues with editing workouts and adding exercises "Editing is hard and sometimes impossible. Adding new exercises also could have been better" (Gamage, 2018).

2.1.5 Summary of Similar applications

Creating a summary in the tables below allows for clear viewing of what features are generally liked or disliked across the range of fitness logging applications reviewed in sections 2.1.1-2.1.4. Using this data some requirement ideas can be formed and questions for potential users can be constructed to get additional information from potential users on what they think about these features.

Table of perceived positive and negative features from the reviewed applications can be found below, in the below table the application numbers represent:

Fitness Builder is 1, Nsuns is 2, HeavySet is 3 and Strong: Exercise Gym Log 5x5 is 4

Application	Positives
1	Ability to share workouts and logged information with personal trainers.
1,3	A wide range of available metrics to measure.
1,2,4	Being able to build or use existing workout routines
2,3,4	Ease of use/Easy data entry
2	Plate calculator
3	Can show weight as a % of 1RM
3	Can import and export data
3,4	Lots of graphs
4,2	Large library of exercises
4	Previous workout data easily visible
4	Warm up calculator

Application	Negatives
1,2	Not much flexibility in being able to swap an exercise
1	Cluttered
1,2	Information and workout data cannot be shared with friends.
1,3	The un-paid version is limited
1	Syncing issues
1,3	Limited exercise selections
2	Limited loggable data
2,4	No online backup
2	Cannot undo incorrect data entry
3	Platform specific application
3,4	Un-intuitive navigation

2.2 Approaches to engaging potential users

In this section different ways of engaging users will be looked at, it's essential to consider all possible approaches to engaging users before setting out a plan of action. This ensures that all the time spent with the users is used in the most optimal way possible, to get as much useful input on the application as feasible.

In section 2.1 records from potential users were inspected, in the form of user reviews, giving indirect user input to inform on desirable features of a workout logging application. Directly engaging with users is another way to get user input, direct observation of the users cannot be done at this stage as a working application would be needed, but could be used at the end in the user evaluation (Kuniavsky, 2003). A posted survey could be performed, but this would have costs associated to it and would rely on the surveys being sent back (Rachelle Seguin, 2004). Cold calling could be done to get user input but this comes with ethical issues of how these phone numbers were acquired. The main methods left to get information from directly engaging with a user are a questionnaire/survey or Interview or Focus Group (Kathy Baxter, 2015).

Interview

Interviews on the surface are an attractive way to gain insight into a subject matter, as they seem to be similar to conversing, however interviews are much more complex and to get the most out of an interview the subject matter you are discussing should also be a complex issue that lends itself to deep discussion (Denscombe, 1998), this can also be time consuming if a fairly large amount of users need to be questioned. This style of data collection will not be appropriate for the purposes of collecting the initial data for this project, as the information needed to be collected doesn't lend itself to complex discussion; and the initially collected data needs to be used to create a set of requirements, which is difficult from an interview style of questioning as all the different responses will need to be interpreted separately. However an interview style of questioning could be employed for the final user evaluation. Furthermore for the final user evaluation 5 users would be appropriate to test the project as they'd be able to find approximately 80% of the usability problems (Nielsen & Landauer, 1993), only interviewing five users will negate issues of having to spend a large amount of time conducting interviews, also as there will be less responses it'll be easier to compare the results of the interviews.

Focus Groups

A focus group is a sub category of an interview, but should not be confused with a group interview; a group interview is where there is still an interviewer that's in charge and asking questions, just like in a regular interview. Where as in a focus group the group discusses a topic between them and the moderator facilitates the group interactions rather than leading the discussion. A focus group is more difficult to set-up than an interview, interviews can be done

online or in person and only require getting together one other person, whereas a focus group needs six to nine willing participants for 1 ½ to 2 hours (Denscombe, 1998). This is not feasible for this project.

Questionnaires/surveys

A questionnaire is most appropriate when it's being used to collect a large number of responses, when the answers tend to be fairly straightforward and uncontroversial (Floyd J. Fowler, 2013). Questionnaire answer data also lends itself to easy analysis as answers can be tallied together. This is appropriate for the initial user data collection for this project as user requirements will be created from the results of this data, so if the preference for the majority of users can be found without having to infer what they're saying, as might be the case with interview style answers, then this is better as we know exactly what the user is saying with their answer.

With a questionnaire it's also important to consider if the questionnaire should be anonymous, and how to structure the questions to avoid useless questions and to keep the users engaged. (Keith G. Diem, 2002)

To summarise what was learnt and also some other pros and cons the table below shows the positives and negatives of each approach. (Denscombe, 1998)

Method	Pros	Cons
Questionnaires /surveys	<ul style="list-style-type: none"> • They can be completed anonymously - if desired. • It's easy to tally answers; don't have to interpret responses. • Can send them out to a lot of people – so more likely to get a lot of viewpoints. • Can be completed online, on paper or verbally. • Can quickly get a lot of data. 	<ul style="list-style-type: none"> • If posted online might not get responses only from the target demographic. • Doesn't allow for probing for more information about why they chose a certain answer/ might not get the full story.
Interview	<ul style="list-style-type: none"> • Get a lot more detailed answers. • The person you're questioning is more likely to be giving you their full attention and really think about their answers. 	<ul style="list-style-type: none"> • Can take a lot of time. • It's harder to compare responses. • Might be influence by interviewer wanting to get certain answers.

Focus Groups	<ul style="list-style-type: none"> • Allow participants to develop ideas between themselves. 	<ul style="list-style-type: none"> • More vocal participants might have their ideas heard a disproportion amount. • Can be hard to get so many people together in one go.
---------------------	---	---

2.3 Information review conclusions

In conclusion, based off the research into various data collection methods, for this project a questionnaire is the most appropriate way to directly question potential users for the initial requirements. This is because it offers a method of data collection that can be sent to a lot of people, doesn't depend on the users being available at a specific time, can be anonymous if desired and it can also be completed online or in person. And the results of the questionnaire lend themselves to becoming requirements of a project as the user answers don't need to be inferred from but simply tallied, this also saves time. There will be no losing out on data as the answers the users can give don't need to be complex for the initial requirements.

However for the final user evaluation an interview style of approach would be better as a smaller number of users will need to be questioned and the questions will lend themselves to more complex discussion.

The investigation into similar applications turned up a lot of interesting positive and negative features of various fitness applications. Showing the features that most applications were praised most often for gives a rough priority of what is essential. Features that were frequently viewed as negative also show what should be avoided. The results of this investigation into similar applications can be used to help build a questionnaire by questioning if the majority of users feel the same way.

3.0 Requirements

This section will put together all the information gathered so far along with the original project objectives to create some requirements for the project that will guide the decisions of what tools to use, design choices, implementation and ultimately aid judgment of if the project was a success.

3.1 Non-Functional Requirements

These requirements are based off what was initially desired for this project.

- Usability
- Extensibility
- Maintainability
- Scalability
- Performance
- Reliability
- Security
- Data Integrity
- Availability
- Flexibility
- Minimalistic

3.2 Requirements from Information Review

These requirements have been collected from the Information review, specifically the summary table (section 2.1.5), the features that appeared numerous times as positives are listed. Also the negative features which appeared multiple times are featured, but have been flipped to become a positive so can also become requirements of the project.

- A wide range of available metrics to measure
- Being able to build or use existing workout routines
- Ease of use/Easy data entry
- Lots of graphs
- Large library of exercises
- Flexibility in being able to swap an exercise
- Information and workout data sharable with friends
- The un-paid version is not limited
- Online backup
- Intuitive navigation
- Easily see past data

3.3 Requirements gathered from questionnaire

The questionnaire used can be found in Appendix A and the results of the questionnaire can be found in Appendix B, the requirements that were created from the user answers are below, these requirements are based on the most sought after features:

- The ability to create workouts or get them from friends or personal trainers
- When selecting a workout routine to be able to:
 - See what other people recommend for certain goals
 - See what body parts are targeted
 - See equipment needed
- They like their workout logging experience to be:
 - Simple
 - Quick to use
 - Intuitive
 - Ability to track lots of data
 - Not having to enter specific data if they don't have it
- The data that should be recordable is:
 - Weight lifted
 - Reps
 - Sets
 - Personal weight
- The Ability to share data should be optional, not mandatory
- When Data should be backed up should be an option to be chosen by the user
- There should be an option to change an exercise as you're working out

3.4 Final Requirements

Whilst building the project the Non-Functional requirements will be kept in mind as a general guidance and have also been used in guiding these final requirements, along with the requirements gathered from the information review and the user questionnaire in addition to the initial objectives taken from Chapter 1 a final set of requirements that the project's success can be judged against can be determined:

- A wide range of metrics to be recorded including but not limited to: Weight lifted, Reps, Sets, Personal weight.
- Data shouldn't have to be compulsory to enter.
- The ability to create workouts.
- The ability to find workouts from other users or friends or personal trainers.
- The ability to be able to log a workout

- Ability to register a personal account with personal details and be able to log into the created account.
- That the application be easy and quick to use, with intuitive navigation and simple data entry.
- Lots of useful graphs are available to see progress – shown on a website, with customisation options.
- A large library of exercises.
- The ability to swap an exercise from a workout.
- The ability to share workouts and workout data if desired.
- An online backup of workout data, and when the data should be backed up should also be customisable.
- It should be easy to see past workout data as you're recording the current workout.

4.0 Development tools and approaches

This section will be investigating many different tools and approaches to software development.

4.1 Available Mobile Platforms

There are many different mobile platforms available to develop for, but given that the percentage of smartphones running iOS or Android natively is 99.6% (Vincent, 2017) and that one of the requirements of this application is to be usable by a majority of people it makes sense to rule anything else out. An exception to this is a mobile web app, which is a website that is developed to look and work in a similar way to a mobile application, so could be run on both Android and iOS (Budiu, 2013). Another option would be a cross platform application, but a primary platform should be chosen to develop on first before developing for cross platform is considered (RapidValue, 2012).

4.1.1 Android

Android currently has a bigger market share than iOS (Teleman, 2017) which means that developing for Android directly would allow the project to be used by more people. Developing for Android does mean that more testing is required however as there are more devices that run Android, with more variation in hardware and screen size, thus increasing the development time (Car, 2015). Android applications can be built on Windows, Linux or Mac machines.

4.1.2 iOS

Developing for iOS is easier to do, this is because you need less lines of code to do the same thing as you would on Android, also across iOS devices the hardware is very similar and the development software is backwards compatible with every iOS device (Hałabuda, 2016). iOS does have a smaller market share than Android, but more importantly than this unfortunately iOS development has to be done on an iOS device (Teleman, 2017). This means developing

with iOS as the primary platform is not possible, as unfortunately for this project access to an iOS device is unavailable, because of this it has to be ruled out due to the hardware limitations.

4.1.3 Mobile web apps

When trying to reach the biggest audience you can a mobile web app is the way to go, this is because it can be run on any device as long as it has access to the internet (Summerfield).

Requiring constant internet access could be a problem though, if we want the application to be usable offline this could not be achieved with a website and if internet signal is patchy you could face difficulties when recording your workout. Also Mobile Web Apps are incapable of sending push notifications or being able to run in the background, in addition to this the requirements state “when the data should be backed up should also be customisable” this could not be met as the data would always have to always be sent as entered.

4.1.4 Conclusion

To conclude, Android is the platform that should be used to develop this application. This is because it has a big market share; the development tools are available on the hardware available for developing this project, and in addition to this a native application allows for many additional features over a mobile web app, both for the current scope of the project - for instance you can run the application in the background and offline. And for future development more options are available, such as push notification and fully offline storage.

4.2 Mobile Development Tools

Since Android has been selected as the platform of choice now possible development tools need to be investigated. For Android there are a lot of editors that use Java, as this is traditionally used to develop for Android, so to represent Java IDE’s Android Studio will be investigated. Cross Platform development tools will be also investigated, one of which is Cordova as it can be used to create Hybrid applications using HTML, CSS and Java. As well as Xamarin which also allows for cross platform development but can build native applications, using C#.

4.2.1 Android Studio

Android Studio is the official IDE (Google) for Android, this means it’s able to build apps on every type of Android device and is always kept up to date with the latest releases of Android and new technology such as wearable devices. It’s also built solely for the purpose of being an Android only development tool meaning it’s optimised for Android and for the deployment of applications to an Android device for testing, which as a result is near instant (Google). No ability to build cross platform applications though limits the potential user base and future development opportunities of the application.

4.2.2 Apache Cordova

Apache Cordova (PhoneGap) is an open source IDE that allows for developing applications for Android, iOS, Windows, Firefox and others all at once. This is because the code base is written in HTML, CSS and JavaScript. It allows offline support, and access to native device APIs. So on the surface seems like a great solution for this project as unlike Android Studio it would allow for future expansion onto other platforms, and it doesn't have the same shortcomings as a mobile web app would. Unfortunately though there are some limitations, for instance the UI for Cordova is shared across platforms, this means that the performance is never as good as it could be when developed for one platform, and Cordova struggles with older devices too where as a native application built in Android Studio wouldn't (Zolciak, 2017).

4.2.3 Xamarin

Xamarin (Microsoft) is another open source IDE that allows for development across iOS, Android and Windows from one core code base, however unlike Cordova or Android Studio this code base is written in C#. Also unlike Cordova a native UI can be built in Xamarin, this gives the performance benefits of Android Studio. But with the option to expand to future platforms by simply having to create the UI for the desired platform and all the backend code can be shared. This also means that any code updates will only need to be done once, improving maintainability. Xamarin is also just as powerful as Android Studio, "Anything you can do in Objective-C, Swift, or Java you can do in C# with Xamarin" (Microsoft).

4.2.4 Conclusion

To conclude, given that our non-function requirements include: Extensibility, Maintainability, Scalability, Performance, Reliability, Data Integrity and Availability. Xamarin is the clear choice as it allows all of these requirements to be hit to a greater degree than Android Studio or Cordova could provide, as it allows for future cross-platform development so increases the availability, scalability, extensibility and maintainability. The performance from an application built in Xamarin is also on par with a natively built application, unlike Cordova.

4.3 Website

A lot of website building is now being done by online tools, where code writing is not required, most web hosting companies now clamour about their website development tools that require no coding experience "Want to build a website without needing to write any code?" (O, 2018). Which is great for people wanting to build specific types of site like ecommerce, but still leave something to be desired when building an application that has specific needs, such as accessing data from an API and showing this data in a chart form. A solution to this is to use Bootstrap (Mark Otto, Jacob Thornton), which also gives you some pre-built templates licence free to build from, is all open source, and allows for you to edit the HTML, CSS and JS yourself and upload the files to any server you like. As for webhosting solutions there are a lot of available

options that for the most part are similar in features and price (Athow, 2018). 1&1 comes with SSL certificates included which meets the security requirement of the project; the service also has very good uptime which meets the reliability and availability requirements. The ability to create and host a database is also included in the price. Chart.js will be used to display graphical content; the library contains 6 charts and is only 11KB making it fast to load and minimalistic.

4.4 APIs

For this project an API makes sense, as some of the same functions will be required on the app and the website, such as the ability to log in. Having one place where this logic happens reduces errors and makes the code more consistent, as only one place has to be updated and tested when required. When updating and inserting data into a database this improves the data integrity as two bits of code won't try to do different things with the same bit of data, once again this also keeps the code more maintainable, and both of these things are part of the non-functional requirements. For this project a simple REST API in PHP will be sufficient.

4.5 Software development approaches

This project lends itself to using an agile development methodology, as there will be a scrum/meeting every two weeks when what has been completed and the objectives for the next two weeks are laid out. This allows for quick defect fixing which is essential when creating an application in a limited amount of time.

4.6 Version Control

This project will need version control setup for the mobile application as well as for the website and API, version control is essential since if a bug is introduced the cause can be found quicker, and it prevents loss of data due to a drive failure. GIT is especially useful when working as a team because of its branching features, however when there is only one creator then these branches can cause some un-necessary hassle as more than likely only one branch will be worked on at a time. SVN however is a solution that is very popular so many clients are available (Sharma, 2015). SVN doesn't feature branching but a single code base, where previous versions and edits of files can quickly be found and merging works well.

4.7 Conclusion

To conclude this project will be built for Android first using Xamarin. There will also be a website hosted on 1&1 with a database also hosted by them along with a PHP REST API. The website and application will use SVN as version control and the whole project will follow an agile development methodology with scrums every two weeks.

5.0 Design

5.1 Mobile app design

Figure 5.1 shows the login screen, it's kept clean and simple to keep with the requirements of the application, with a big login button to allow for easy pressing. Nothing else is required on this screen so nothing else should be included.

Figure 5.2 shows the main menu screen, this screen will be shown after you click the login button. The buttons are big so they're easy to press and the text makes it clear what screen each button will take you to.

The design for the login screen is titled "Workout Logger". It features two input fields: "Name" and "Password". Below these fields is a large "Login" button.

Figure 5.1: Login Screen Design

The design for the main menu screen is a vertical list of buttons: "Log Workout", "Create Workout", "Search for workout", "Modify current workout", "Personal Data Logging", and "Settings".

Figure 5.2: Main Menu Screen Design

Figure 5.3 shows the layout design for when you're logging a workout, this screen is the one that the user will probably spend the most time on, and will be using whilst exercising. So it's important that the buttons on this screen are big enough to be easily clickable, even when sweaty and exhausted, yet also need to leave enough room for valuable information to be shown clearly. To do this an expandable /dynamic layout will be used, to make the most of all available screen space. Also once an exercise is completed it should disappear off the screen to save room. This ensures that the requirements "the application be easy and quick to use, with intuitive navigation and simple data entry" and "be easy to see past workout data as you're recording the current workout" can both be met.

The design for the log workout screen is titled "Day x". It features a table with columns for "Reps", "Weight", and "Rest Time (s)", and an "Add set" button. Below the table are three rows of "x" values. At the bottom, there are three buttons: "Skip Exercise", "Timer", and "Complete". Below these are two rows of "Remaining Exercise" with "Jump To" buttons.

Day x			
Exercise Name			
Reps	Weight	Rest Time (s)	Add set
x	x	x	
x	x	x	
x	x	x	
Skip Exercise		Timer	Complete
Remaining Exercise 1		Jump To	
Remaining Exercise 2		Jump To	

Figure 5.3: Log Workout Screen Design

Figure 5.4 shows the personal data screen, this is so users can record data related to their bodies. The last filled in data will be visible as a hint. There will be a large amount of possible fields and they don't all have to be filled in to submit the data. This fulfils the "A wide range of metrics to be recorded" requirement.

Figure 5.4: Personal Data Screen Design

Figure 5.5: Modify/Create workout Screen Design

Also not all data needs to be filled in meeting the requirement that "Data shouldn't have to be compulsory to enter."

Figure 5.5 shows the screen for modifying and creating a workout, these screens should be identical. The only difference being the modify workout screen will be pre-populated by the current workout or one of the workouts that can be searched for and then edited. This fulfils the requirement for "The ability to create workouts, or find them from other users or friends or personal trainers."

Figure 5.6 is a very basic screen all that is required to search for existing workouts is the workout name, the workouts will then appear below that with additional information about the workout such as the user that made it and the name of the days in the workout if available.

Figure 5.6: Search for Workout Screen Design

Figure 5.7: Settings Screen Design

Figure 5.7 shows the settings screen, many options should be visible here to fulfil the requirements about customisation.

5.2 Database Design

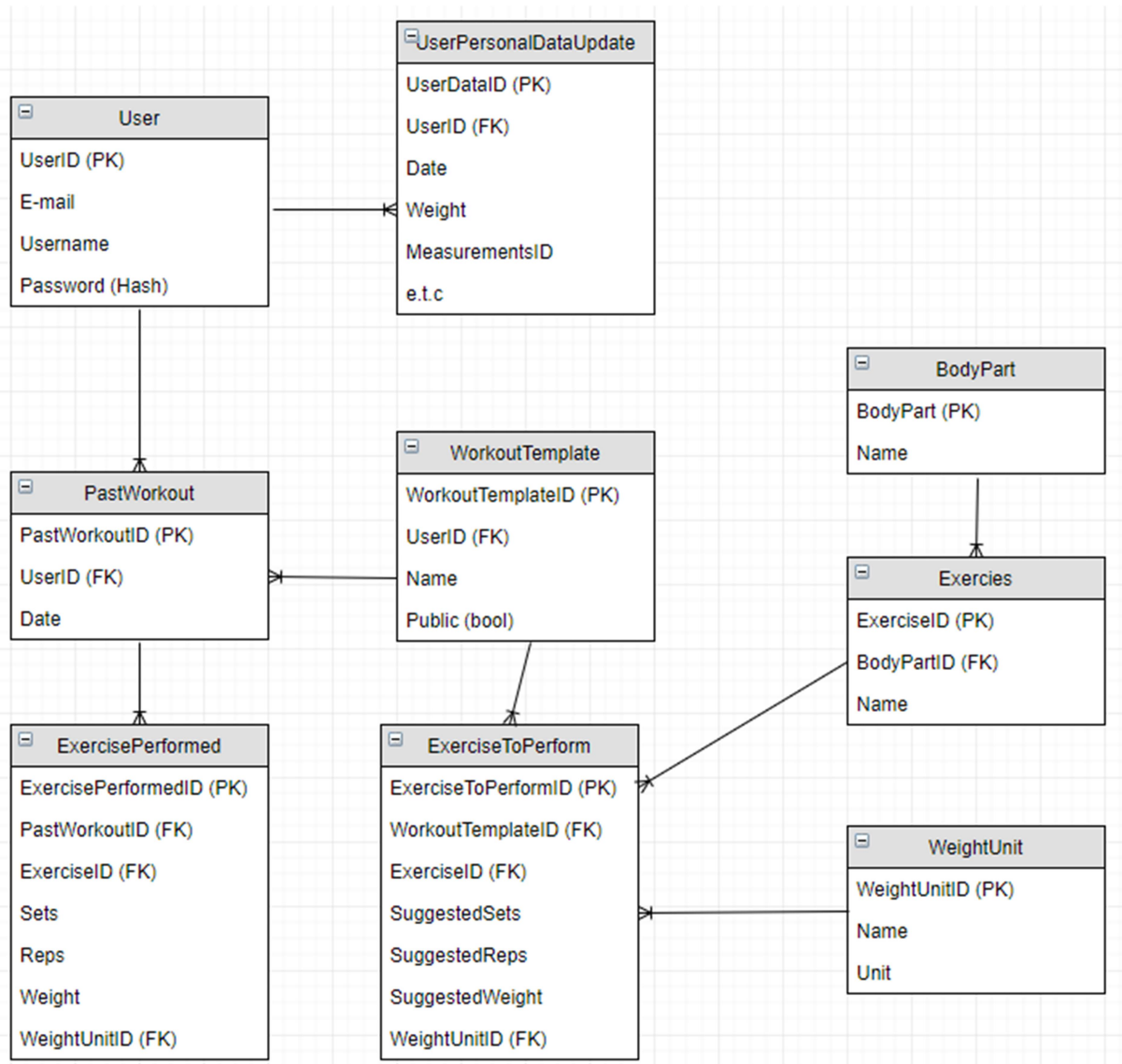


Figure 5.8: Initial database design

Figure 5.8 is the initial database design, this design will evolve as more scenarios/ certain aspects of the program are developed. But this initial design on the surface allows the requirements to be met and is a good starting point for the database.

5.3 Website Design



Figure 5.9: Main website design screen

The website can be simple in nature as seen in figure 5.9; all it needs is the ability to be able to show data in an informative and useful way to the users. It should also have some customisation options, hence the settings page.

In addition to this the users will be required to log in, so a login dialog/popup will be required as show in figure 5.10. If there is the ability to login there should also be the ability to register an account and recover a forgotten password, as well as being able to logout.

The image shows a login dialog form. It has two input fields: 'E-mail' with the text 'Example@Example.com' and 'Password' with a masked password '*****'. Below the password field is a checkbox labeled 'Remember password'. A large 'Login' button is positioned below the checkbox. At the bottom of the dialog, there are two links: 'Register an account' and 'Forgotten Password'.

Figure 5.10: Login dialogue

5.4 API Design

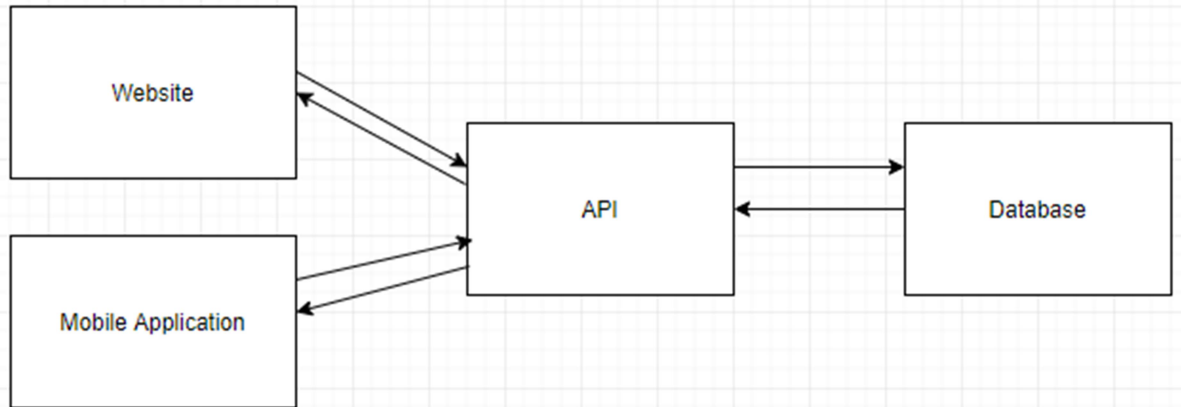


Figure 5.11: API design

The API works as a middle man to manage all post and get requests from the website and mobile app as shown in figure 5.11, this ensures that the returned data and how data is inserted into the database is consistent between all applications, thus increasing the reliability and maintainability as well as reducing the amount of testing required.

An API contract, which is a piece of technical documentation on how to create and use an API (Sturgeon, 2017), doesn't need to be made available to the users of this application as all API calls are currently for internal use only. However consideration into what will be required of the API and the input/output of each call should be considered, more methods may be added in development; but the methods shown below will all be required.

Method	Input	Output
Post Register	First name, last name, E-mail, password	Success or failure
Post Login	Email, Password	UserToken
Get Exercises by name	Search term	Json Data { basic exercise data }
Get Specific Exercise Data	ExercisePK	Json Data { full exercise data }
Post Created Workout Data	Json Data { full created workout data }, UserToken	Success or failure
Get Current Workout Day data	UserToken	Json Data { workout day data }
Get Current workout Day with previous data	UserToken	Json Data { workout day data with past values }
Post Completed Workout Data	User Token, Json Data { Completed workout data }	Success or failure
Get Workout Templates	UserToken, SearchTerm	Json Data { basic workout template }

		data}
Get Specific Workout Template	UserToken, WorkoutTemplatePK	Json Data {workout template}
Get Personal Data	UserToken	Json Data {user data}
Post Personal Data	UserToken, Json Data {user data}	Success or failure

6.0 Implementation

6.1 Database

The original database was constructed from the database design in section 5.2, this served as a basic implementation to help development of other aspects of the project. One of the first decisions to be made was over how the passwords should be stored. This application doesn't contain too much personal/confidential information but security is still a concern, passwords should not be stored as plain text so SHA-256 will be used to hash the passwords, it's more secure than MD5 which is considered "cryptographically broken and unsuitable for further use" (The CMU Software Engineering Institute, 2009), but isn't over the top for this application.

It was also realised that passing the user password around all the time would be bad for security, as even when sent over a secure connection potential vulnerabilities could expose the posted data, and an attacker accessing a user token wouldn't be nearly as bad as a password which the user could have re-used on other sites. So to implement tokens a "LoginTokens" table was added, this table is populated on login with a userID, a randomly generated token (64 characters in length), and a time stamp. This token can then be used to verify a user for up to two hours after the token was generated. This token should always be passed to the API when any API call is made.

The tables "Advanced Sets" and "Completed Sets" needed adding to allow for an exercise to have advanced details for more than one set, the "Advanced Sets" table was used for the initially created exercises and "Completed Sets" used for when workout data is recorded for these sets. The records in "Advanced Sets" and "Completed Sets" have a many to one relationship with the "Exercise to Perform" and "Exercise Performed" tables respectively.

The Table "Day" was also added, this table was needed to store the individual days of a Workout Template. Some fields and relationships were also changed to reflect these changes, and to accommodate for any other required features of the application. These final changes can be seen in figure 6.1 (on the next page).

Many stored procedures were also added in the database, for the most part these stored procedures match up with to an API function that calls it, but a brief explanation of these

procedures can be found in Appendix G, along with some testing that was carried out on the stored procedures during development.

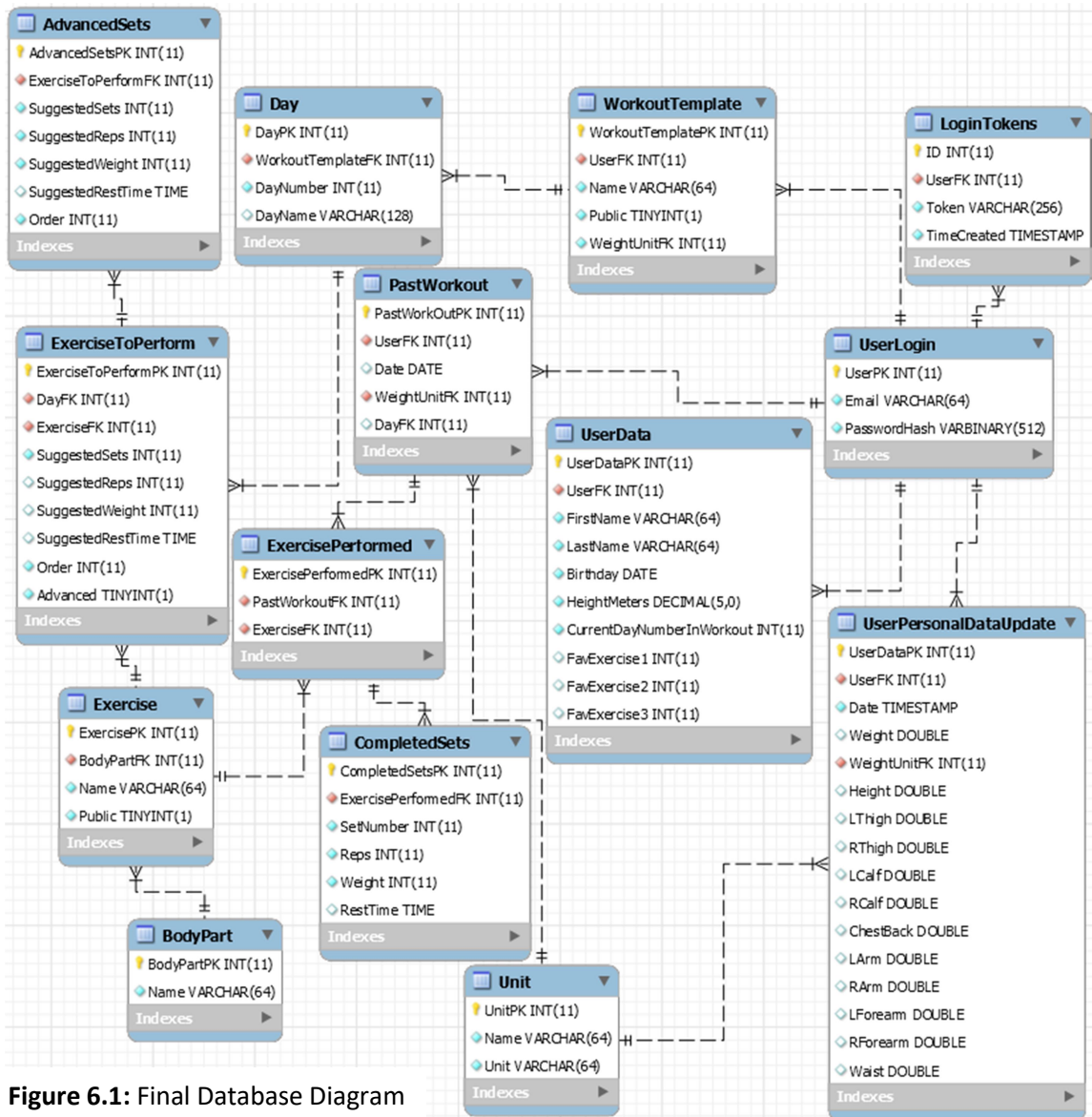


Figure 6.1: Final Database Diagram

6.2 API

Credit for the initial design/template used to help build this API is found on Mark Roland's website who is a professional web developer (Roland, 2012). This template is excellent as it makes it very easy to see how to implement authentication and to set if a HTTPS connection is required, both of which could improve the security of the application. You can also provide a parameter to tell the API how you'd like the response, as plain text, XML or JSON.

Where possible to save having to repeatedly publish the API and to improve maintainability stored procedures were used.

An example of this can be seen in figure 6.2, using stored

procedures makes the API code very simple and maintainable. Sometimes though a stored procedure cannot be used, for instance when submitting a completed workout, a few options present themselves, this could be split into multiple API calls for individual bits of data, but then problems could arise if connection is interrupted with partially submitted data. Or a SQL query could be built in

application and then sent to the API, but then more data will be sent to the API and could potentially leave the API exposed if any SQL can be sent to it and executed blindly. So instead the alternative picked as shown in figure 6.3, was to instead pass the workout

```
case 'getCurrentWorkoutDayData' :
    $userToken = $_POST['token'];
    $sql = "CALL sp_GetWorkoutDayData('$userToken')";
    $result = $conn->query($sql);

    $response['code'] = 1;
    $response['status'] = $api_response_code[ $response['code'] ]['HTTP Response'];
    $jsonData = array();

    if ($result->num_rows > 0) {
        while($row = $result->fetch_assoc()) {
            $jsonData[] = $row;
        }
    } else {
        echo "0 Results";
    }

    $response['data'] = $jsonData;

break;
```

Figure 6.2: Code snippet of get Current Workout Day Data API procedure.

```
case 'postCompletedWorkoutData' :
    $workoutData = $_POST['workoutData'];
    $userToken = $_POST['token'];

    $obj = json_decode($workoutData, true);

    $dayNumber = $obj['Day'];

    $buildingInsertString = "SET @userID = (SELECT UserID FROM LoginTokens WHERE Token = '$userToken');
    INSERT INTO PastWorkout (UserFK, Date, WeightUnitFK, DayFK) VALUES (@userID, NOW(), 0, $dayNumber);
    SET @PastWorkOutPK = LAST_INSERT_ID();";

    foreach ($obj['ListOfWorkoutExercises'] as $ListOfWorkoutExercises) {
        $exerciseNumber = $ListOfWorkoutExercises['ID'];
        $exerciseOrder = $ListOfWorkoutExercises['Order'];
        $exerciseSets = $ListOfWorkoutExercises['Sets'];
        $exerciseReps = $ListOfWorkoutExercises['Reps'];
        $exerciseAdvanced = $ListOfWorkoutExercises['Advanced'];
        $exerciseAdvancedString = ($exerciseAdvanced) ? 'true' : 'false';
        $exerciseRestTime = $ListOfWorkoutExercises['RestTime'];

        if ($exerciseNumber == null) {
            $exerciseName = $ListOfWorkoutExercises['Name'];
            $buildingInsertString = $buildingInsertString . "INSERT INTO Exercise (Name) VALUES ('$exerciseName');
            SET @ExercisePK = LAST_INSERT_ID();";
            $exerciseNumber = "@ExercisePK";
        }

        $buildingInsertString = $buildingInsertString . "INSERT INTO ExercisePerformed ('ExerciseFK', 'PastWorkoutFK')
        VALUES ($exerciseNumber, @PastWorkOutPK);
        SET @ExercisePerformedPK = LAST_INSERT_ID();";

        foreach ($ListOfWorkoutExercises['SubSetList'] as $subSet) {
            $subSetNumber = $subSet['SubSet'];
            $subSetReps = $subSet['SubReps'];
            $subSetWeight = $subSet['SubWeight'];
            $subSetRestTime = $subSet['RestTime'];

            $buildingInsertString = $buildingInsertString . "INSERT INTO CompletedSets
            ('ExercisePerformedFK', 'SetNumber', 'Reps', 'Weight', 'RestTime') VALUES
            (@ExercisePerformedPK, $subSetNumber, $subSetReps, $subSetWeight, '$subSetRestTime');";
        }
    }

    $buildingInsertString = $buildingInsertString . "CALL sp_UserNextWorkoutDay(@userID);";
    $buildingInsertString = $buildingInsertString . "SELECT @PastWorkOutPK;";

    $result = $conn->multi_query($buildingInsertString);
```

Figure 6.3: Code snippet of post Completed Workout Data API procedure.

data object and the user token to the API and construct the SQL inside the API, this makes the code a bit harder to maintain, but is the best option available.

In addition to the API methods that were mentioned in the design of the API, in Section 5.4 API Design, API methods were also added to get past workout data in table format for the website and a method to submit the favourites selected on the website settings page. Two API methods were also required by the application to get the Workout Day Names to display on the settings page and another API call to set the workout day if the user changes this setting.

6.3 Mobile Application

The first challenge of the mobile application was to get a login system working, as a user account will be required at every other stage of the application to get user specific information. Because of the API all that is required is to put the submitted E-mail and Password into a post request and send it to the API with the login method name, the user token will then also be returned from the API.

A lot of the backend of the application is done by the API, because of this it helps keep the application simple, small in size and fast. Where the main thing the application does is display and post data. This is good as it allows UI to be the main focus of the application, so a lot of thought can be put into making the UI as easy to use as possible, which is what the user base requested, and is needed for an application that will be used in the gym when tired and sweaty as it makes reading and pressing small buttons difficult. The API calls are very quick as the sent and received data is deliberately kept to the smallest size possible. These calls are so fast that for the exercise name on each key press the API is queried to get possible suggestions and this works very smoothly.

With this in mind it was elected that where possible the layout should be dynamic, this is more of a time commitment to get set-up, but will make the most of all possible space on the screen. This can be seen in figure 6.4 to figure 6.5, when the advanced option is not checked the previously added exercises are visible, however when advanced is selected and

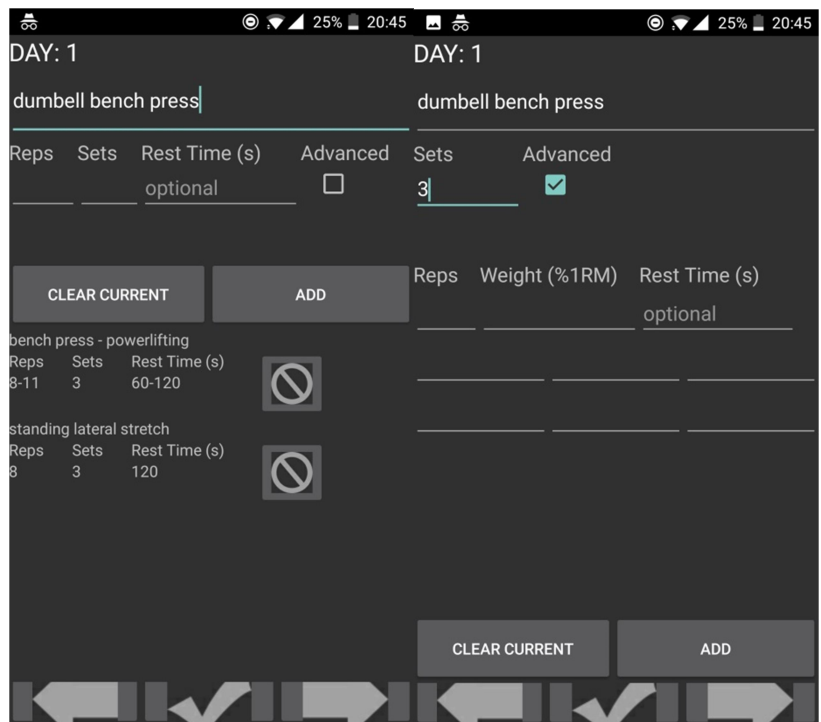


Figure 6.4: Create Workout Screen

Figure 6.5: Create Workout Screen Advanced Checked

more editable text boxes are needed, then the previously added exercises are hidden, this prevent the screen from getting crowded. In addition to this the layout will expand to fill as much space as it can, but will also shrink when required, for instance when the keyboard is shown, as seen in figure 6.6. Although this makes the buttons a bit smaller they're still usable and at this point aren't the main focus, the user will be focused on entering data on the keypad at this point, where the numbers are more than big enough.

Figure 6.7 and 6.8 show the log workout screen, this is the screen that will be used most often and used in the gym so it's important for the design to be well thought out. The buttons are all large with spaces between each of them. When clicked the timer expands as the buttons next to it will not be needed when the timer is active. The bottom navigation bar doesn't exist on this page as instead once all exercises are completed it

will be assumed the workout is completed so the workout will auto-complete, this was also done to save space on a busy screen where space should be used in the most efficient ways possible. The past data for the current exercise is easily visible as hints in the boxes where the new data will be entered and the exercises coming up at the bottom also have a quick insight into the number of sets, reps and rest times of each exercise.

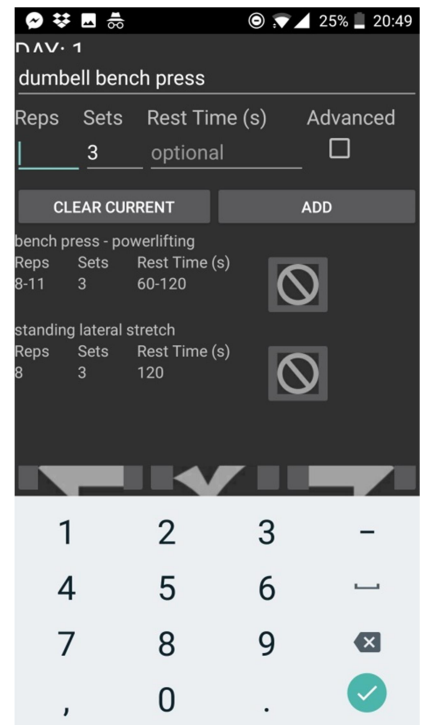


Figure 6.6: Create Workout Screen shrunk to allow keyboard

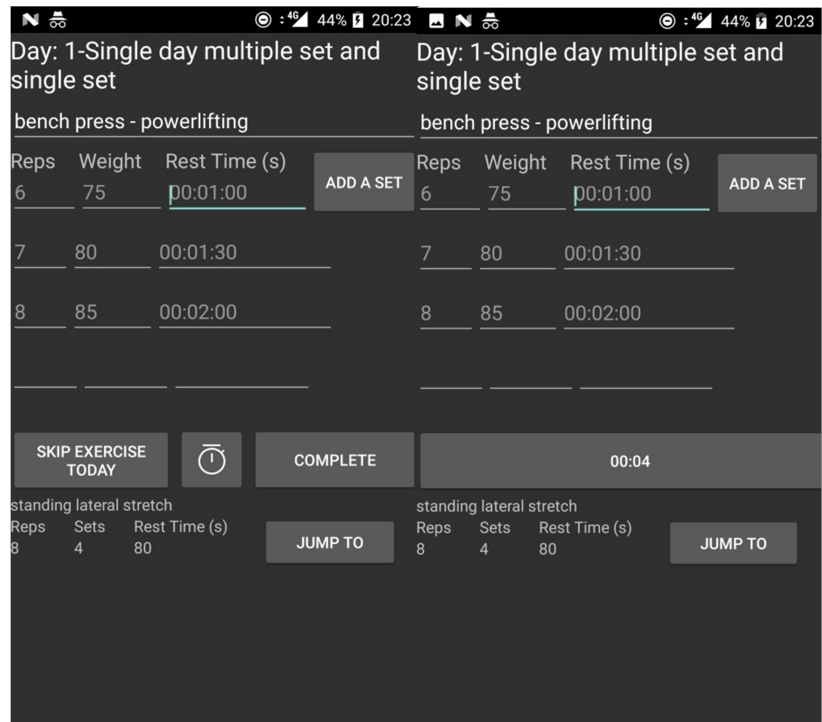


Figure 6.7: Log workout before timer pressed

Figure 6.8: Log workout after timer pressed

The Personal data logging screen is shown in Figure 6.9, this screen contains a lot of options for recordable personal data, far exceeding the number of available measurements set out in the requirements. But one of the aims of this project was to be able to “track a vast amount of data related to fitness”, which this screen allows, the detailed measurements section is scrollable but the main measurements will always appear at the top. Each part of the personal data is only sent to the API if it is updated. This keeps the sent data small and as the data is saved like a log with the current date and time this creates detailed historical records that allow for detailed data analysis, as set out in the aims and requirements.

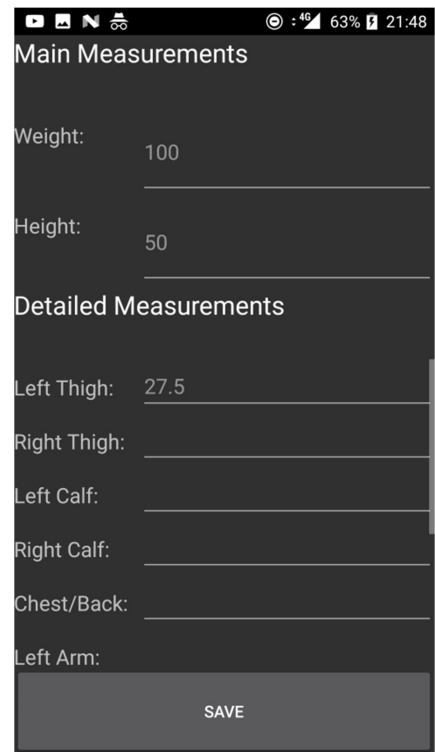


Figure 6.9: Personal details to log.

The settings page shown in figure 6.10 contains an option to change the day, unfortunately due to time constraints some of the desired settings were not added, such as when to upload the workout data. These settings would have to be added in future implementations.

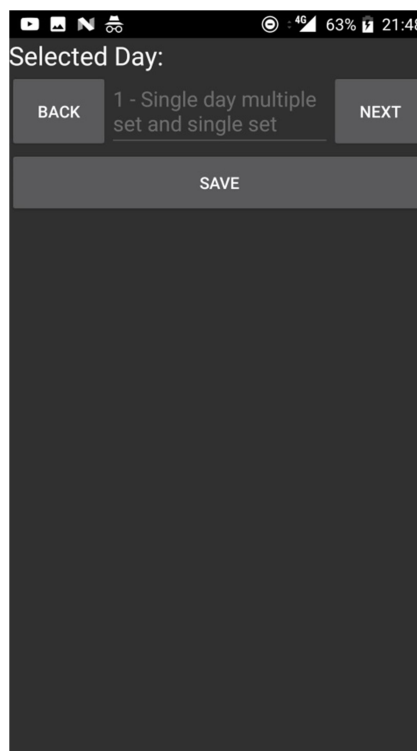


Figure 6.10: Settings

The search workout page, shown in figure 6.11 shows the workout name, the number of days in the workout and the names of the workout days if available.

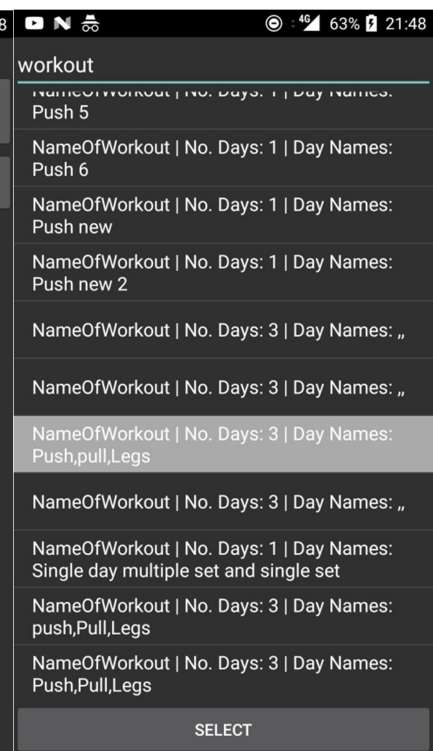


Figure 6.11: Search for workouts

There is no unique identifier though; this should be added for users wanting to find specific workouts. Also more search options would make this page more user friendly. Again these things weren't added because of time constraints.

6.4 Website

The bootstrap template used for the website is called SB Admin (Blackrock Digital); this template comes free for modification and distribution under the MIT Licence. This template is great as it uses Chart.js, has pages set up for login, registration, forgotten password and a dashboard. It also has a simple clean layout and will need little adjusting to make it ideal for this project.

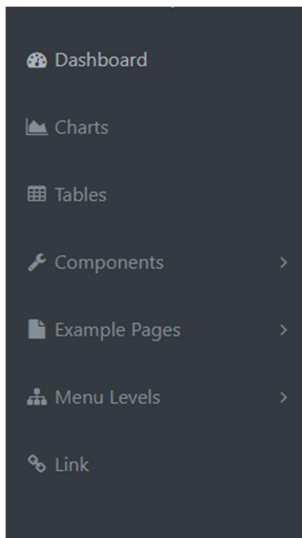


Figure 6.7: Original SB Admin template navigation

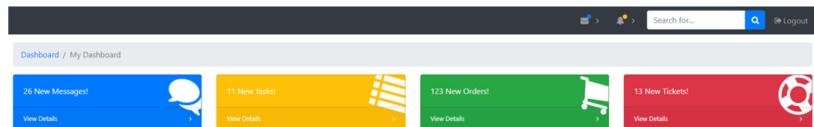


Figure 6.6: Original SB Admin template

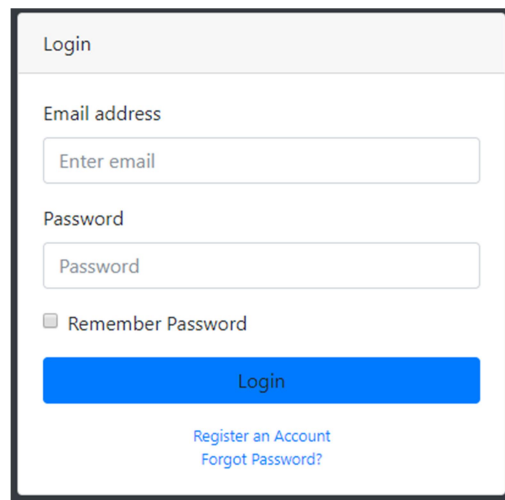


Figure 6.8: Original SB Admin template for login

At this stage this project has no use for searching, alerts and messages, which can be seen in figure 6.6. So these will have to be removed. In addition to this the navigation menu as seen in figure 6.7 contains more options than needed for this project, some have to be removed and the others should be re-named.

Figure 6.8 shows the login screen, this login screen is excellent and very similar to the designs for a login that were made in chapter 5.3, this can be kept as it is and just change the login button to call the API login. The same can be said for the registration page but will instead just need to send the registration data to the API.

All of the data fetching and sending is done through the API. When the returned data for the graphs is returned it is looped through and using JavaScript with PHP code it's put inside an array to be passed to the graphs.

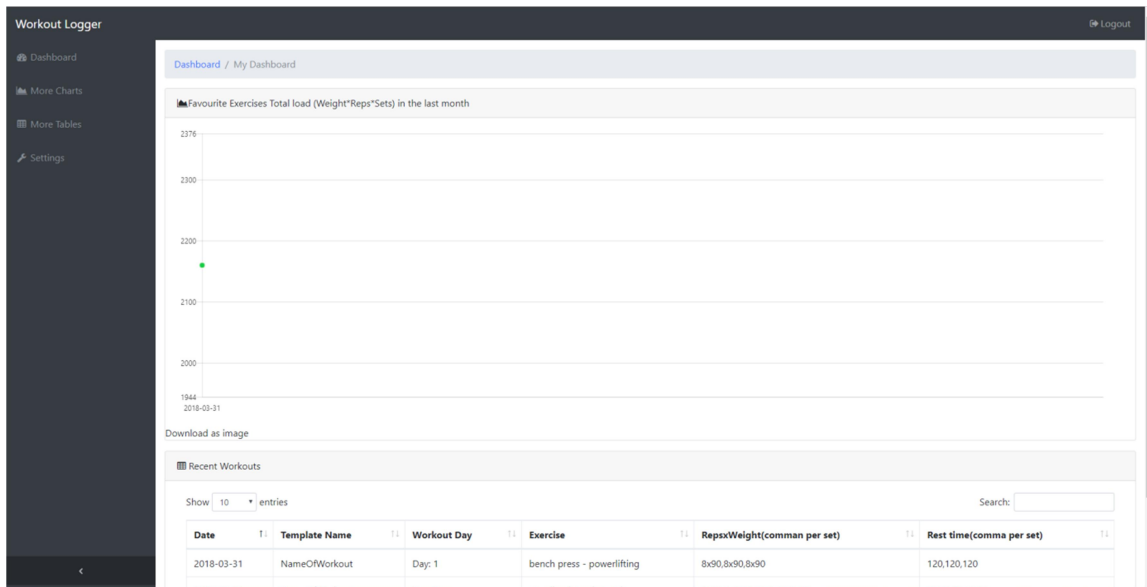


Figure 6.9: Completed dashboard of Workout Logger

Figure 6.9 shows the finished Dashboard, this graph has only been populated by one data point, so doesn't show much information right now. Under the graph is an option to save the graph as an image, this could then be shared with whoever you like, or simply save locally if desired. Under this is a table which shows the most recent workout data for this user. The bar at the top has been tidied up to just contain a button for logging out, the left hand navigation bar has also had the workout logger title added and the options are now relevant to this application. There is also the option at the bottom to shrink the navigation bar; this is good for on smaller mobile screens to make the most of available space so there is more space for the content.

The settings page has the option to change the user favourite exercises as seen in figure 6.10. When these exercises are changed it changes the exercises shown on the dashboard graph. The selected exercise names can be seen from the dashboard by hovering over the data points. The Charts page shows a graph for personal data but that is the only other graph on the website right now.

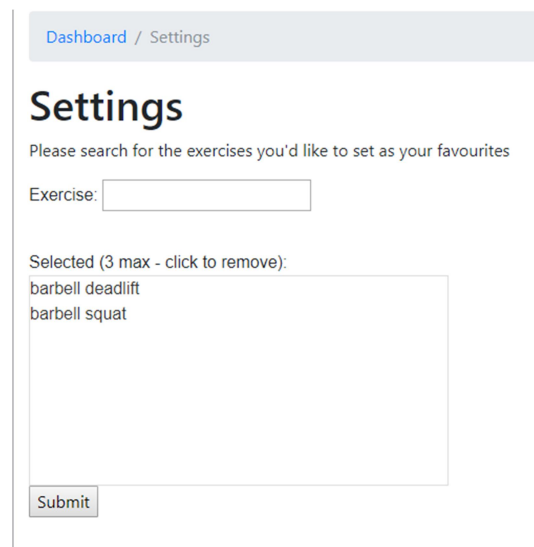


Figure 6.10: Website settings page

6.6 Local Testing

Testing was done throughout the development as part of the agile development process. This helped keep the project relatively bug free. At the end of development some usability testing was carried out to check that all features were working as expected. A full table of these tests and their results can be seen below.

Test	Expected Result	Actual Result
1 - Create user account	User login and detail records are created after registering user	The records were successfully inserted into both the UserData and UserLogin tables
2 - Logging into a user account on the website	Successful login	Login worked successfully, all data loaded as expected.
3 - Attempt to login to website using invalid details	Login un-successful	As expected login was not allowed, no error shown but login wasn't successful
4 - Logging into a user account on the mobile application	Successful login	Login worked successfully, all data loaded as expected.
5 - Attempt to login to mobile application using invalid details	Login un-successful	As expected login was not allowed, no error shown but login wasn't successful
6 - Search for exercise by name	Suggestions to be presented in real-time as typing	Suggestions are displayed as expected in a timely manner
7 - Create workout on mobile application with an advanced set exercise and non-advanced exercise and multiple days	Single record inserted into Workout Template, multiple records in day, multiple records for the day records in exerciseToPerform and Advanced sets for the advanced exercises in AdvancedSets	Single record inserted into Workout Template, multiple records in day, multiple records for the day records in exerciseToPerform and Advanced sets for the advanced exercises in AdvancedSets. All with the correct details – worked as expected.
8 – Loading workout into logging screen	Loading a newly created workout should load the first day of that workout with all expected exercises and suggested sets and reps.	All data was correctly loaded as expected
9 – Adding sets to an	Clicking the add set button should	Text boxes were correctly created

exercise	create new text boxes to be filled out	
10 – Filling out and submitting workout data	Text boxes should be editable, but values shouldn't be required in every text box to submit data	Text boxes were editable, restricted to appropriate input. Not all text boxes had to be filled in for successful data submission.
11 - Skip Exercise	Skip exercise button should remove the current exercise and load the next	Worked as expected
12 – View submitted data on website in table	Submitted data should be viewable on the dashboard page when signed into a user account	Data was visible and correct
13 – Change workout day	Settings page should allow workout day to be changed	Day was changed and the selected day appears when going onto the logging workout screen
14 – Recording personal data	Any amount of entered data should be able to be submitted and on returning to the personal data screen the latest data should be visible	Data could be submitted and viewed as expected.
15 - Viewing past workout data on logging screen	Data from the previous week/last time the exercise was performed by this user should be visible	Data was visible as a hint in text boxes and was all correct
16 – Searching for other workouts	Workouts can be searched for by name and selected to become the active workout routine	Workout was correctly found and assigned to user when selected
17 – Selecting favourite exercises for website	Exercises should be able to be searched for and selected as favourites.	Exercises could be searched for, selected and removed
18 – Checking favourite exercise volume chart	Chart should be populated with correct data when a new entry is submitted.	Data was visible as expected
19 – Personal weight graph	On submitting a weight update the graph should be updated with this weight	Weight was visible on the graph on the correct day as expected
20 – Ability to search for and add new exercises	When searched for if an exercise can be found it should be selectable, otherwise submitting	Exercises could be found and created as expected.

	just an exercise name without selecting an existing one will create a new exercise	
21 – Modify workout	Workout should be modifiable and when changed the changes will be saved correctly	Workout was changeable, including being able to remove and add a new exercise as well as changing recommended sets and reps. Changes were then saved.
22 – Saving Graph data	Graph data should be saveable	Graph data is saveable, the background is black though which makes the chart harder to read than on the website.

8.0 Evaluation

8.1 User Evaluation

Five of the people that took the initial questionnaire expressed that they would be happy to take part in an interview and to test the product after the project was completed. As discussed in Section 2.2 this is ideal for this user evaluation section, as this part lends itself to an interview format, as more detailed answers can be given and more follow up questions can be asked, five participants for UX testing is also ideal.

A rough interview guide can be found in Appendix C, these questions were used to keep the interview on track and make sure the interview went in a productive direction. In Appendix D is the notes taken from each interview; a list summarising the feedback given in the interviews is in the following user feedback section and numbered so they can be referenced in the reflection of requirements section.

8.1.1 User Feedback

The feedback in Appendix D is in the note form it was taken as whilst interviewing the users after they'd used the app and website for a minimum time of 5 minutes, notes were also taken from verbal feedback and observation as they navigated the applications. The applications were then kept open for them to reference back to whilst the interview was under way. These notes have been put into a list of points with any very similar points being merged into one, and slightly re-written in places, this list is below and is sorted into generally positive, neutral or negative groups:

Positives

1. Likes the data backup and how you can see past measurements and workouts in text boxes
2. Application was simple, quick and easy to use
3. The navigation made sense
4. Data Entry was easy
5. A lot of available exercises
6. Likes the dynamic layout, in particular the expanding timer, screen space is well used, making things easy to see and buttons easy to press
7. Likes how many measurements can be taken
8. Likes how you can leave some of the logging fields blank
9. Likes the past workout table
10. Likes how workouts are shared
11. Likes how you save the graph data
12. The website looks nice, liked the big graphs, how they look and customisation options

Neutral

13. Simple colours and text, bland but easy to read
14. Likes the website on laptop but it doesn't work as well on their phone

Negative

15. Unclear on what the % meant in relation to weights
16. No units given for weights or personal measurements
17. Up button on modify workout screen unclear also modify current workout and modify someone else's workout screens are slightly different
18. Advanced checkbox unclear until pressed
19. No calorie counting options or cardio specific options
20. No mention of if should record bar weight
21. Why limit to 3 exercises on the favourites graph also wishes there were more graphs
22. Would like a calculator for what weights to put on the bar

8.1.2 Summary

Generally the responses were positive, some problems were found though. Quite a few of the problems related to certain things not being clarified in the interface and/or an assumed level of knowledge from the user, such as in points 15,16, 17, 18 and 20. The users all found the application usable still, and were able to navigate and accomplish tasks such as creating and completing a workout and then viewing this data.

Points 19, 21 and 22 were all requests for features the users felt were missing. With such a large application meeting every desire from every user is impractical, but these are features that could be looked into in future developments.

The majority of feedback was positive though with the simplicity of the application and the data entry plus the data backup, ease of past data viewing and the graphs amongst the things that were liked the most.

8.2 Reflection of Requirements

The tests done under usability testing (section 6.6) and the summary of the user evaluation (section 8.1.1) points are referenced and used as proof through this table.

Requirement	Proof of meeting the requirement
A wide range of metrics to be recorded including but not limited to: Weight lifted, Reps, Sets, Personal weight.	There are a wide range of metrics that can be recorded, including the ones listed in this requirement. Figure 6.7 and 6.9 in the mobile application implementation showed this working, as did the tests 10 and 14. This requirement has been met
Data shouldn't have to be compulsory to enter.	Data is not compulsory to enter, entire exercises can be skipped, or just single sets. The rest time also doesn't need to be entered. This is shown as working in tests 10 and 11. This requirement has been met
The ability to create workouts	Figure 6.4 shows a workout being created. The creation is fully implemented proof of this is in text 7. So this requirement has been met.
The ability to find workouts from other users or friends or personal trainers.	Figure 6.11 shows a workout being searched for, proof of this working is in test 16; the search could do with more work though, as there is no unique identifier for a workout visible and only one thing that can be searched on. So this requirement has been mostly met.
The ability to log a workout	This has been fully implemented, proof of which can be found in test 10. So this requirement has been met.
Ability to register a personal account with personal details and be able to log into the created account.	This has been fully implemented, proof of which can be seen in tests 1 and 2. So this requirement has been fully met.
That the application be easy and quick to use, with intuitive navigation and simple data entry.	Proof of meeting this requirement can be seen in the user evaluation under points 2, 3 and 4.
Lots of useful graphs are available to see progress – shown on a website, with customisation options.	Multiple useful graphs and tables were created, the proof of which can be seen in test 18, 19 and 12 and the user evaluation shown 12. There are also customisation options available, shown in test 17. More graphs and tables could be added. But this requirement is met.
A large library of exercises.	A large library of exercises were imported and more

	can be added, as shown in test 20 and in the user evaluation 5. So this requirement is met.
The ability to swap an exercise from a workout.	Exercises can be swapped out of an exercise. As shown in test 21. So this requirement is met.
The ability to share workouts and workout data if desired.	Workouts are sharable, and graph data is exportable as an image so can also be shared if desired as shown in tests 16 and 22, however it cannot be selected as to what the workout should be named or if you'd like to share it or not. So this requirement has been partially met.
An online backup of workout data, and when the data should be backed up should also be customisable.	Online data is backed up when submitted as shown in test 12, unfortunately due to time restraints it cannot be selected by users when this data should be backed up. So this requirement has only been partially met.
It should be easy to see past workout data as you're recording the current workout.	Can be seen in figure's 6.7 and 6.9, proof of this is mentioned in user evaluation 1. So this requirement has been met.

8.3 Reflection of aims

The very initial aims of this project as found in Appendix E were to:

- I aim to provide a sophisticated yet on the surface a very simple way to track a vast amount of data related to fitness.
- And to provide a way to analyse this data in great detail so you can find out what works best for any given user so they can reach their fitness goals.
- I want this information to optionally be sharable and for users to be able to choose what data they want to share, so you can share it on social media or with a personal trainer.

Although these aims were met as shown in section 8.1.1 user feedback points 2, 4, 7, 11 and 12, along with the tests in section 6.6 tests 10, 12,17 and 22. The initial aims were slightly more orientated towards analysing the data that is collected by the application. More could have been done to truly fulfil and surpass these aims.

8.4 Summary

The initial aims of this project were all met, and all the local tests were a success. However this doesn't mean that the project is perfect, this is shown in figure 8.1 with the user feedback, the majority of the comments (55%) were positive and the application is usable, but there are a number of areas that could be improved on with both clarifications in the UI and some additional features.

As for the requirements that were laid out in the final requirements (Section 3.4) figure 8.2 shows that 77% of these requirements were fully implemented with the remaining 23% being partially implemented. The partially implemented requirements could be met by allowing users to select if they'd like to share their workout routine and allowing more advanced search options when searching for workout routines. Also when logged workout data should be uploaded should be customisable. However neither of these things hinders the usability of the application.

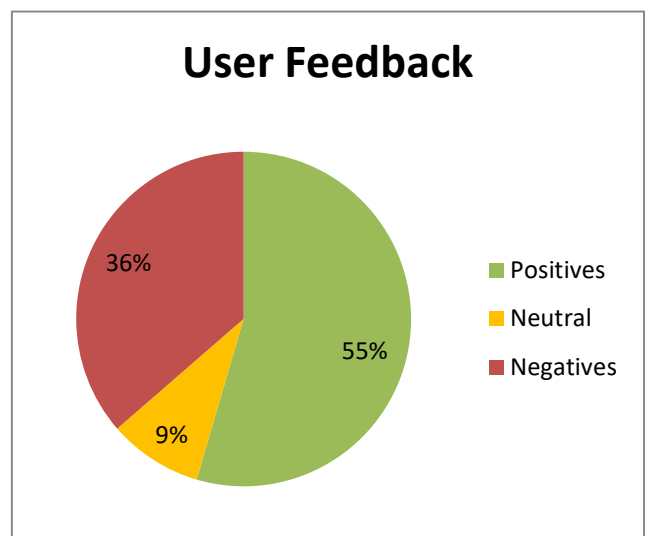


Figure 8.1: Pie Chart of user feedback

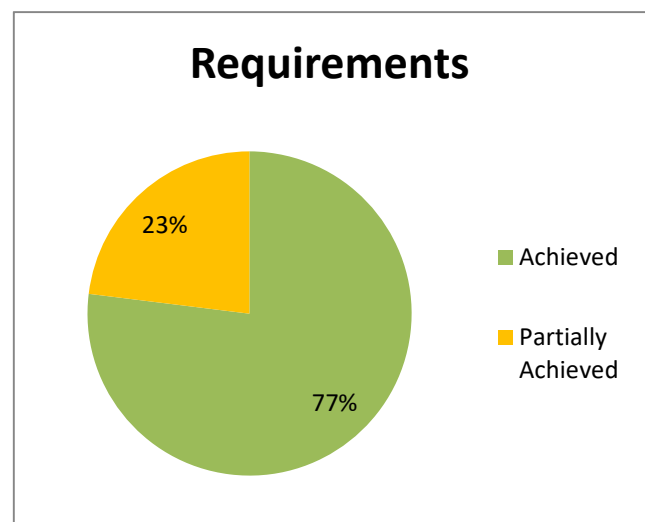


Figure 8.2: Pie Chart of requirements

8.5 Future Work

There are many areas that lend themselves for future development, for instance the partially completed requirements, specifically the ability to select if your workout routine is available to search for, opposed to it being automatically shared, the database has the capabilities for this so it'd be very feasible to implement into the application. Along with being able to customise when your workout data should be uploaded, such as only when connected to W-Fi, when at home e.t.c, to accomplish this device storage would need utilising more. This would also mean the application uses the internet less, but would take up slightly more space on the device. More search options would also be good on the search for other workouts pages, to allow users to filter what workouts they want to search for.

Also user feedback on clarifications that need making in the UI, such as what the % means in the weight section when using a new workout with an advanced exercise. And the units need adding for weights and measurements; this unit should also be customisable, which is set up in the database but not in the application. An additional requested feature in the user feedback was for a plate calculator, this could also be added.

In addition to these points getting a security certificate for the website so that https can be used for the API calls would make the application more secure as this prevents the users against man in the middle attacks where information being sent to a website can be stolen. This is essential to protect the user passwords on the initial login before the user has a token.

9.0 Critical Review

The objectives of this project were quite ambitious; there were so many elements to the project which meant I had a lot to learn, this was a positive experience though. I managed to meet the main aims of the project that I set out to meet and even most of the requirements, the local tests of the actual code were also all a success and the problems that the users had were primarily minor UI issues that had been overlooked and some additional features they desired after seeing the application. I think this shows that the agile style of development and testing was a success along with the fact that it allowed for the functionality of the project to remain mostly bug free. So I think that my approach was well suited to meet the aims of such a big project. And that I managed my time well throughout; my initial deadlines, found in Appendix E that I set before I began my project were all met on time and helped to keep me on schedule.

However the end result of the project feels a little underwhelming, there are so many aspects to the project, so it doesn't focus specifically on one small element. So no individual aspect is done to an exceptional standard as doing this would compromise time available to spend getting the whole project to a working level. This project lends itself to a great deal of future development. But with time this base application could become an industry quality application. With this said perhaps if I was to do this project again I'd focus more on one smaller element rather than trying to build many different aspects of a large application. But I do think this project was a good exercise into how a multifaceted application should be developed.

At the start I had to consider ethics, you can see the consideration of ethics in Appendix E and F. No ethical concerns were identified as the users should be able to select any and all data that can be shared. However as it cannot be selected in the current version of the project if a workout routine is shared or not, this creates minor ethical issues as a personal trainer may not want their private workout routines to be shared without being paid. This is an issue that would have to be addressed before public release.

The ethics of the questionnaire and interview also had to be considered. The participants of each were given the option to be completely anonymous, to be able to leave at any time, and made to feel as comfortable as possible with the emphasis being that the application was being tested, not them, and that there were no right or wrong answers.

References

- Anonymous. (2015, July 12). *FitnessBuilder*. Retrieved March 27, 2018, from Google Play:
https://play.google.com/store/apps/details?id=com.pumpone.fitnessbuilder&reviewId=Z3A6QU9xcFRPRIVfSnFiT3M1elhONG1RNGp1N0dObkk4S1JfS3pqM2thYzhMZEtFa3U2T25Ca2p1c1ZnUk03Yk90SnRPdEF1ZHJLOHdvVkpCM2NZckpwX0pn&hl=en_GB
- Athow, D. (2018, March 28). *The best web hosting services for 2018*. Retrieved March 29, 2018, from techradar: <https://www.techradar.com/news/best-web-hosting-services>
- B, E. (2018, January 14). *Nsuns 5/3/1 Program Log*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=com.monstrapps.nsun531program&reviewId=Z3A6QU9xcFRPRjBZRDNVU2plbG1rWXZqVGZoQ3V6N3FjWF9PcGxhSzgyV3R3ZnNncWRxV3pUeS1KOVb6cnBwT2ZKZVlhVUdVeVFyQjZGeXR5WVBOaVRkZGc>
- Blackrock Digital. (n.d.). *SB Admin*. Retrieved March 31, 2018, from Start Bootstrap:
<https://startbootstrap.com/template-overviews/sb-admin/>
- Budiu, R. (2013, September 14). *Mobile: Native Apps, Web Apps, and Hybrid Apps*. Retrieved March 28, 2018, from nngroup: <https://www.nngroup.com/articles/mobile-native-apps/>
- Car, T. (2015, October 27). *Android development is 30% more expensive than iOS. And we have the numbers to prove it!* Retrieved April 9, 2018, from infinum: <https://infinum.co/the-capsized-eight/android-development-is-30-percent-more-expensive-than-ios>
- Denscombe, M. (1998). *The Good Research Guide For Small-scale social research projects Fourth Edition*. Berkshire: Open University Press.
- Depwe, T. (2014, October 9). *Fitness Builder*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=com.pumpone.fitnessbuilder&reviewId=Z3A6QU9xcFRPRXR0R0MyUIFjVXFrQTBKcIRIMnBuSUt5U09KVXFDRGNYZIJxSjZHakJVvW5pQXZkdFRfMnRGWk43c3pwRmx5RDFrRFVrSTISS0FmMHBOVEdF>
- Floyd J. Fowler, J. (2013). *Survey research methods*. Sage publications.
- Gamage, R. (2018, March 22). *Strong: Exercise Gym Log, 5x5*. Retrieved March 2018, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9xcFRPRUo4UVZ1dFIFcU56Vmp6U01CNnNZTEt6VTVuZ0pmeWJZeXBTNXpZWXJSTWsxYTVBNXfDbTBMWEs3M3RTVm1sMUNhMG5Cb0pOajJQM1FQWWNn>

- Google. (n.d.). *Android Studio*. Retrieved March 29, 2018, from Android:
<https://developer.android.com/studio/index.html>
- Hałabuda, P. (2016, March 24). *Mobile Application Development: iOS vs Android vs Windows Phone*. Retrieved March 28, 2018, from WhallaLabs: <http://whallalabs.com/mobile-application-development-ios-vs-android-vs-windows-phone/>
- Kathy Baxter, C. C. (2015). *Understanding Your Users: A Practical Guide to User Research Methods*. Morgan Kaufmann.
- Keith G. Diem, P. (2002). *A Step-by-Step Guide to Developing Effective Questionnaires and Survey Procedures for Program Evaluation & Research*. New Brunswick: Rutgers-Cook College Resource Center.
- Kuniavsky, M. (2003). *Observing the User Experience: A Practitioner's Guide to User Research*. Elsevier.
- lokkheed. (2018, March 27). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9xcFRPRmFpVXlkaEYwUDBLVUV4Z000OHdsX3cybDhCeHNpYzdabF9VQIJTVIBzaFkyeURXRTdZcTNTSDRhV0lReGhXUmgwbVlxUW8xdHllUEtXeEdR>
- Mark Otto, Jacob Thornton. (n.d.). *Bootstrap*. Retrieved March 29, 2018, from [getbootstrap](https://getbootstrap.com/):
<https://getbootstrap.com/>
- Microsoft. (n.d.). *Xamarin*. Retrieved March 29, 2018, from [Xamarin](https://www.xamarin.com/):
<https://www.xamarin.com/>
- MonstrApps. (2017, November 29). *Nsuns 5/3/1 Program Log*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=com.monstrapps.nsun531program&hl=en>
- Nielsen, J., & Landauer, T. K. (1993). A mathematical model of the finding of usability problems. *ACM INTERCHI'93 Conference*, (pp. 206-213). Amsterdam.
- O, J. (2018, February 7). *Choose a website builder: 17 top tools*. Retrieved March 29, 2018, from Creative Blog: <https://www.creativebloq.com/web-design/best-website-builder-8133804>
- Pérez, M. C. (2017, November 10). *Nsuns 5/3/1 Program Log*. Retrieved March 27, 2018, from Google Play:

<https://play.google.com/store/apps/details?id=com.monstrapps.nuns531program&reviewId=Z3A6QU9xcFRPRXZzQUpb2pCTngzZfVXZWR5Vm9TQ2FQclpqcXo3ZXh0cHdNSUp0MTITUkJCNTyMVINRU9OWWcxdu1wRnBCZjZvWXhpcFdwMmt6YTYdyMnc>

PhoneGap. (n.d.). *Cordova*. Retrieved March 29, 2018, from Cordova:

<https://cordova.apache.org/>

PumpOne. (2017, December 20). *FitnessBuilder*. Retrieved March 27, 2018, from Google Play:

https://play.google.com/store/apps/details?id=com.pumpone.fitnessbuilder&hl=en_GB

Rachelle Seguin, M. G. (2004). *E-mail or snail mail? Randomized controlled trial on which works better for surveys*. Canadian Family Physician.

RapidValue. (2012). *How to Choose the Right Architecture*.

Roland, M. (2012, November 14). *PHP API Template*. Retrieved March 31, 2018, from Mark

Roland: <http://markroland.com/portfolio/restful-php-api>

Runloop Ltd. (2018, March 15). *HeavySet Strength Training Log*. Retrieved March 27, 2018,

from iTunes: <https://itunes.apple.com/us/app/heavysset-gym-log-1rm-tracker/id1171500310?mt=8>

Saunders, N. (2018, March 27). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play:

<https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9xcFRPSEZON2RXMG9NS013X290a3gzaHR2Wm9meV9lXzdackRjTzdqOENTbXh1azVPQkh6dS1lVfphOHZKTXBjc1ZSaGVKNkVPVzRra2dEU1ZBNXkw>

SH, R. (2014, August 6). *FitnessBuilder*. Retrieved March 27, 2018, from Google Play:

<https://play.google.com/store/apps/details?id=com.pumpone.fitnessbuilder&reviewId=Z3A6QU9xcFRPRk1wWINSWjlFN2ZTeFJsaDB2aE1qbHFEWDJHYVdOQmYxQjM5emt1OTRYX0VJMUF5RFR5VnpvSUoxM0VxVmZKU1VZRF9HeGR6UW1KRUZNbXJr>

Sharma, R. (2015, February 18). *Top 5 open source version control tools for system admins*.

Retrieved March 29, 2018, from getfilecloud:

<https://www.getfilecloud.com/blog/2015/02/top-5-open-source-version-control-tools-for-system-admins/#.Wrz9aIjwaUk>

Snow, A. (2014, August 3). *Fitness Builder*. Retrieved March 27, 2018, from Google Play:

<https://play.google.com/store/apps/details?id=com.pumpone.fitnessbuilder&reviewId=Z3A6QU9xcFRPSDQ4TEV0MF9kVjROU0cWdEFGRmVsY0xUbK9iR0VpYjVnWC1lc>

TVxNmXKNWQ1MEVklXFGXzNZLXd6OUtuS1RNbTNxWVhUcHRha0UtQkF3X1
Zv&hl=en_GB

Strong Fitness. (2018, March 26). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play: <https://play.google.com/store/apps/details?id=io.strongapp.strong>

Sturgeon, P. (2017, September 15). *Commit to API Contracts*. Retrieved April 4, 2018, from APIs You Won't Hate: <https://blog.apisyouwonthate.com/guessing-api-contracts-ac1b7eabced>

Summerfield, J. (n.d.). *Mobile Website vs. Mobile App: Which is Best for Your Organization?* Retrieved March 29, 2018, from Human Service Solutions: <https://www.hswsolutions.com/services/mobile-web-development/mobile-website-vs-apps/>

Teleman, C. (2017, March 3). *Android vs. iOS App Development: A Comparison and Decision Guide [Infographic]*. Retrieved March 28, 2018, from dzone: <https://dzone.com/articles/android-vs-ios-app-development-a-comparison-and-de>

The CMU Software Engineering Institute. (2009, January 21). *Vulnerability Note VU#836068 MD5 vulnerable to collision attacks*. Retrieved March 31, 2018, from Vulnerability Notes Database: <https://www.kb.cert.org/vuls/id/836068>

Vincent, J. (2017, February 16). *99.6 percent of new smartphones run Android or iOS*. Retrieved March 28, 2018, from The Verge: <https://www.theverge.com/2017/2/16/14634656/android-ios-market-share-blackberry-2016>

Waddell, J. (2018, March 22). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play: <https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9xcFRPRkRYVE51bmRaRVJja3FSSnI2UklBcW14bTIJZjBJcmJwOHNQdFJuZ3lXRUFvTUxMa3Uzck5mYlhCUEdLUUF4aUdmTWdUN2tQR01yVnlPOEhR>

Zolciak, A. (2017, March 16). *Why is Xamarin Vs. Cordova the Wrong Question to Ask?* Retrieved March 29, 2018, from InsaneLab: <https://insanelab.com/blog/mobile-development/xamarin-vs-cordova-cross-platform-development/>

Bibliography

- Anonymous. (2015, July 12). *FitnessBuilder*. Retrieved March 27, 2018, from Google Play:
https://play.google.com/store/apps/details?id=com.pumpone.fitnessbuilder&reviewId=Z3A6QU9xcFRPRIVfSnFiT3M1elhONG1RNGp1N0dObkk4S1JfS3pqM2thYzhMZEtFa3U2T25Ca2p1c1ZnUk03Yk90SnRPdEF1ZHJLOHdvVkpCM2NZcKpwX0pn&hl=en_GB
- Athow, D. (2018, March 28). *The best web hosting services for 2018*. Retrieved March 29, 2018, from techradar: <https://www.techradar.com/news/best-web-hosting-services>
- Australasia, V. E. (Director). (2011). *Data : Ethical use and storage* [Motion Picture].
- B, E. (2018, January 14). *Nsuns 5/3/1 Program Log*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=com.monstrapps.nsun531program&reviewId=Z3A6QU9xcFRPRjBZRDNVU2plbG1rWXZqVGZoQ3V6N3FjWF9PcGxhSzgyV3R3ZnNncWRxV3pUeS1KOVb6cnBwT2ZKZVlhVUdVeVFyQjZGeXR5WVBOaVRkZGc>
- Blackrock Digital. (n.d.). *SB Admin*. Retrieved March 31, 2018, from Start Bootstrap:
<https://startbootstrap.com/template-overviews/sb-admin/>
- Budiu, R. (2013, September 14). *Mobile: Native Apps, Web Apps, and Hybrid Apps*. Retrieved March 28, 2018, from nngroup: <https://www.nngroup.com/articles/mobile-native-apps/>
- Car, T. (2015, October 27). *Android development is 30% more expensive than iOS. And we have the numbers to prove it!* Retrieved April 9, 2018, from infinum: <https://infinum.co/the-capsized-eight/android-development-is-30-percent-more-expensive-than-ios>
- Denscombe, M. (1998). *The Good Research Guide For Small-scale social research projects Fourth Edition*. Berkshire: Open University Press.
- Depwe, T. (2014, October 9). *Fitness Builder*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=com.pumpone.fitnessbuilder&reviewId=Z3A6QU9xcFRPRXR0R0MyUIFjVXFrQTBKcIRIMnBuSUt5U09KVXFDRGNYZIJxSjZHakJVvW5pQXZkdFRfMnRGWk43c3pwRmx5RDFrRFVrSTISS0FmMHBOVEdF>
- Floyd J. Fowler, J. (2013). *Survey research methods*. Sage publications.
- Gamage, R. (2018, March 22). *Strong: Exercise Gym Log, 5x5*. Retrieved March 2018, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9>

xcFRPRUo4UVZ1dFIFcU56Vmp6U01CNnNZTEt6VTVuZ0pmeWJZeXBTNXpZWX
JSTWsxYTVBNXFDbTBMWEs3M3RTVm1sMUNhMG5Cb0pOajJQM1FQWWNn

Google. (n.d.). *Android Studio*. Retrieved March 29, 2018, from Android:
<https://developer.android.com/studio/index.html>

Hałabuda, P. (2016, March 24). *Mobile Application Development: iOS vs Android vs Windows Phone*. Retrieved March 28, 2018, from WhallaLabs: <http://whallalabs.com/mobile-application-development-ios-vs-android-vs-windows-phone/>

Hellerstein, J. M. (2005). *Readings in Database Systems*. The MIT Press.

Hermes, D. (2015). *Xamarin mobile application development : Cross-platform C# and Xamarin.Forms fundamentals*. California: Apress.

Kathy Baxter, C. C. (2015). *Understanding Your Users: A Practical Guide to User Research Methods*. Morgan Kaufmann.

Keith G. Diem, P. (2002). *A Step-by-Step Guide to Developing Effective Questionnaires and Survey Procedures for Program Evaluation & Research*. New Brunswick: Rutgers-Cook College Resource Center.

Kuniavsky, M. (2003). *Observing the User Experience: A Practitioner's Guide to User Research*. Elsevier.

lokkheed. (2018, March 27). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9xcFRPRmFpVXlkaEYwUDBLVUV4Z000OHdsX3cybDhCeHNpYzdabF9VQIJTVIBzaFkyeURXRTdZcTNTSDRhV0lReGhXUmgwbVlxUW8xdHllUEtxeEdR>

Mark Otto, Jacob Thornton. (n.d.). *Bootstrap*. Retrieved March 29, 2018, from getbootstrap:
<https://getbootstrap.com/>

Microsoft. (n.d.). *Visual Studio Documentation*. Retrieved April 8, 2018, from Microsoft:
<https://docs.microsoft.com/en-us/visualstudio/>

Microsoft. (n.d.). *Xamarin*. Retrieved March 29, 2018, from Xamarin:
<https://www.xamarin.com/>

Microsoft. (n.d.). *Xamarin Documentation*. Retrieved April 8, 2018, from Microsoft:
<https://docs.microsoft.com/en-us/xamarin/>

- MonstrApps. (2017, November 29). *Nsuns 5/3/1 Program Log*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=com.monstrapps.nsun531program&hl=en>
- Nielsen, J., & Landauer, T. K. (1993). A mathematical model of the finding of usability problems. *ACM INTERCHI'93 Conference*, (pp. 206-213). Amsterdam.
- Nixon, R. (2009). *Learning PHP, MySQL & JavaScript: With JQuery, CSS & HTML5*. Shroff Publishers & Distributors.
- O, J. (2018, February 7). *Choose a website builder: 17 top tools*. Retrieved March 29, 2018, from Creative Blog: <https://www.creativebloq.com/web-design/best-website-builder-8133804>
- Oliver, P. (2010). *Student's Guide to Research Ethics*. Maidenhead: McGraw-Hill Education.
- Pérez, M. C. (2017, November 10). *Nsuns 5/3/1 Program Log*. Retrieved March 27, 2018, from Google Play:
<https://play.google.com/store/apps/details?id=com.monstrapps.nsun531program&reviewId=Z3A6QU9xcFRPRXZzQUpnb2pCTngzZGVXZWV5Vm9TQ2FQclpqcXo3ZXh0cHdNSUp0MTITUKJCNTryMVINRU9OWWcxduU1wRnBCZjZvWXhpcFdwMmt6YTYdyMnc>
- PhoneGap. (n.d.). *Cordova*. Retrieved March 29, 2018, from Cordova:
<https://cordova.apache.org/>
- PumpOne. (2017, December 20). *FitnessBuilder*. Retrieved March 27, 2018, from Google Play:
https://play.google.com/store/apps/details?id=com.pumphone.fitnessbuilder&hl=en_GB
- Rachelle Seguin, M. G. (2004). *E-mail or snail mail? Randomized controlled trial on which works better for surveys*. Canadian Family Physician.
- RapidValue. (2012). *How to Choose the Right Architecture*.
- Richardson, L. (2013). *RESTful Web APIs*. O'Reilly.
- Ritscher, W. (Director). (2016). *Xamarin Essential Training* [Motion Picture].
- Roland, M. (2012, November 14). *PHP API Template*. Retrieved March 31, 2018, from Mark Roland: <http://markroland.com/portfolio/restful-php-api>

- Runloop Ltd. (2018, March 15). *HeavySet Strength Training Log*. Retrieved March 27, 2018, from iTunes: <https://itunes.apple.com/us/app/heavysset-gym-log-1rm-tracker/id1171500310?mt=8>
- Saunders, N. (2018, March 27). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play: <https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9xcFRPSEZON2RXMG9NS013X290a3gzaHR2Wm9meV9lXzdackRjTzdqOENTbXh1azVPQkh6dS1lVFphOHZKTXBjc1ZSaGVKNkVPVzRra2dEU1ZBNXkw>
- SH, R. (2014, August 6). *FitnessBuilder*. Retrieved March 27, 2018, from Google Play: <https://play.google.com/store/apps/details?id=com.pumphone.fitnessbuilder&reviewId=Z3A6QU9xcFRPRk1wWINSWjlFN2ZTeFJsaDB2aE1qbHFEWDJHYVdOQmYxQjM5emt1OTRYX0VJMUF5RFR5VnpvSUoxM0VxVmZKU1VZRF9HeGR6UW1KRUZNbXJr>
- Sharma, R. (2015, February 18). *Top 5 open source version control tools for system admins*. Retrieved March 29, 2018, from getfilecloud: <https://www.getfilecloud.com/blog/2015/02/top-5-open-source-version-control-tools-for-system-admins/#.Wrz9aIjwaUk>
- Snow, A. (2014, August 3). *Fitness Builder*. Retrieved March 27, 2018, from Google Play: https://play.google.com/store/apps/details?id=com.pumphone.fitnessbuilder&reviewId=Z3A6QU9xcFRPSDQ4TEV0MF9kVjROU0cWdEFGRmVsY0xUbk9iR0VpYjVnWC1lcTVxNmXKNWQ1MEVklXFGXzNZLXd6OUtuS1RNbTNxWVhUcHRha0UtQkF3X1Zv&hl=en_GB
- Strong Fitness. (2018, March 26). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play: <https://play.google.com/store/apps/details?id=io.strongapp.strong>
- Sturgeon, P. (2017, September 15). *Commit to API Contracts*. Retrieved April 4, 2018, from APIs You Won't Hate: <https://blog.apisyouwonthate.com/guessing-api-contracts-ac1b7eaeabcd>
- Summerfield, J. (n.d.). *Mobile Website vs. Mobile App: Which is Best for Your Organization?* Retrieved March 29, 2018, from Human Service Solutions: <https://www.hswsolutions.com/services/mobile-web-development/mobile-website-vs-apps/>
- Teleman, C. (2017, March 3). *Android vs. iOS App Development: A Comparison and Decision Guide [Infographic]*. Retrieved March 28, 2018, from dzone: <https://dzone.com/articles/android-vs-ios-app-development-a-comparison-and-de>

The CMU Software Engineering Institute. (2009, January 21). *Vulnerability Note VU#836068 MD5 vulnerable to collision attacks*. Retrieved March 31, 2018, from Vulnerability Notes Database: <https://www.kb.cert.org/vuls/id/836068>

Vincent, J. (2017, February 16). *99.6 percent of new smartphones run Android or iOS*. Retrieved March 28, 2018, from The Verge: <https://www.theverge.com/2017/2/16/14634656/android-ios-market-share-blackberry-2016>

Waddell, J. (2018, March 22). *Strong: Exercise Gym Log, 5x5*. Retrieved March 27, 2018, from Google Play: <https://play.google.com/store/apps/details?id=io.strongapp.strong&reviewId=Z3A6QU9xcFRPRkRYVE51bmRaRVJja3FSSnI2UklBcW14bTIJZjBJcmJwOHNQdFJuZ3lXRUFvTUxMa3Uzck5mYlhCUEdLUUF4aUdmTWdUN2tQR01yVnlPOEhR>

West, A. W. (2013). *Practical PHP and MySQL website databases : A simplified approach*. Berkeley: Apress.

Zolciak, A. (2017, March 16). *Why is Xamarin Vs. Cordova the Wrong Question to Ask?* Retrieved March 29, 2018, from InsaneLab: <https://insanelab.com/blog/mobile-development/xamarin-vs-cordova-cross-platform-development/>

Appendices

Appendix A - Questionnaire

Questionnaire that was designed is available on the following three pages.

Fitness logging application - weightlifting focused questionnaire

Fitness logging application - weightlifting focused questionnaire. This data will be collected and used to inform decisions on creating a fitness related application. All responses will be anonymous.

1. **Have you ever made your own workout routine?**

Mark only one oval.

- Yes
 No

2. **If you just answered no would you consider it?**

Mark only one oval.

- Yes
 No
 Maybe

3. **Do you have a workout routine you follow?**

Mark only one oval.

- Yes
 No *Skip to question 7.*

Workout routine questions

4. **Where did you get your current workout routine from?**

Mark only one oval.

- I created it
 Personal trainer
 Found it online
 Other: _____

5. **What did you look for when choosing or creating your current workout routine?**

Tick all that apply.

- Workout length
 Body parts targeted
 Equipment needed
 Flexibility of the workout
 How fun the workout looked
 I followed others guidance
 Other: _____

6. **If you created it how did you create it?**

Mark only one oval.

- Looking up exercises myself and writing them down
- A dedicated app
- A dedicated website
- Other: _____

7. **How do you track your workouts?**

Mark only one oval.

- I don't *Skip to question 13.*
- Pen and paper
- Note taking application
- Dedicated mobile app
- Other: _____

Tracking your workout questions

8. **What do you like about this method of tracking workouts?**

Tick all that apply.

- It's simple
- It's quick to use
- It's intuitive
- You can track a lot of data
- Other: _____

9. **What do you dislike about this method of tracking workouts?**

Tick all that apply.

- It's slow
- It's hard to use
- It needs specific things entered
- It's over complicated
- Other: _____

10. **What data do you track? Please check all that apply.**

Tick all that apply.

- Weight lifted
- Reps
- Sets
- Time taken
- Difficulty
- Total session time
- Your personal weight
- Your body measurements
- Calorie intake
- Macro Nutrition data (protein, fats, carbs)
- Other: _____

11. **Would you like the option to be able to share the data you track?**

Mark only one oval.

- Yes - I want to share it on social media and with personal trainers
- Yes - just with personal trainers/people I trust though
- I'm unsure
- No
- Other: _____

12. **When would you like to upload your fitness data?**

Mark only one oval.

- As you enter it
- At the end of the workout
- Whenever I'm connected to Wi-Fi
- When I choose
- No preference
- Other: _____

Final question

13. **Do you ever have to change what exercise you're meant to be doing due to lack of equipment e.t.c?**

Mark only one oval.

- Yes
- No
- Not sure/maybe

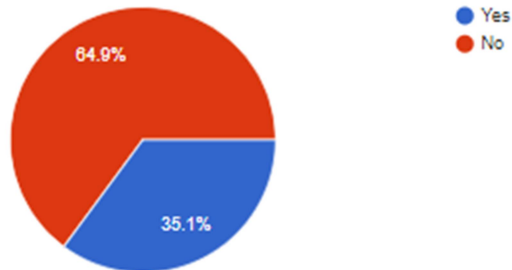
Appendix B – Questionnaire Results

Questionnaire results on the next 6 pages

Have you ever made your own workout routine?

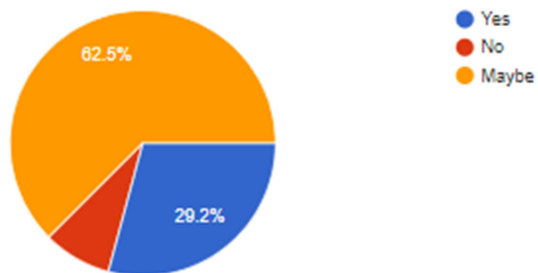


37 responses



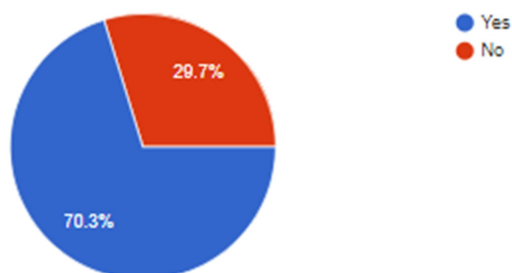
If you just answered no would you consider it?

24 responses



Do you have a workout routine you follow?

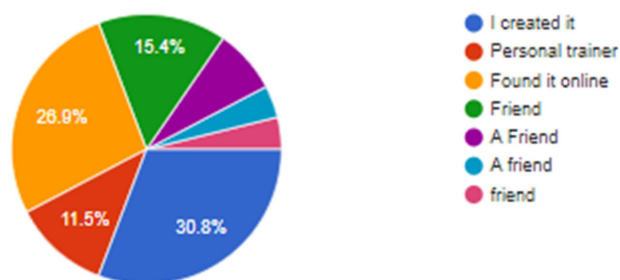
37 responses



Workout routine questions

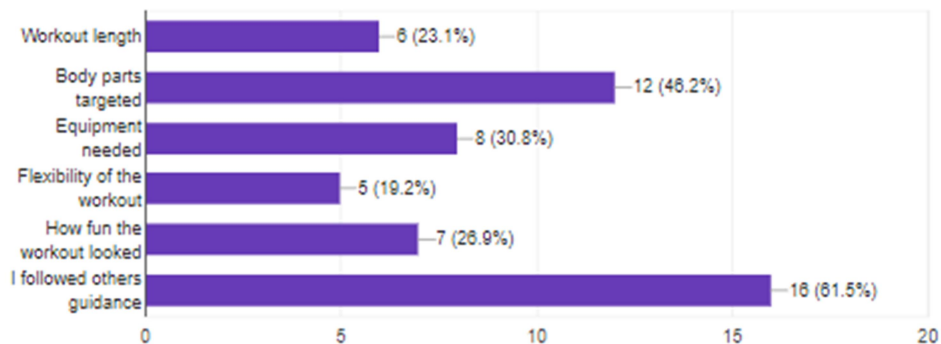
Where did you get your current workout routine from?

26 responses



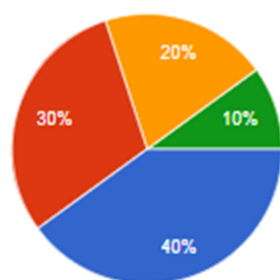
What did you look for when choosing or creating your current workout routine?

26 responses



If you created it how did you create it?

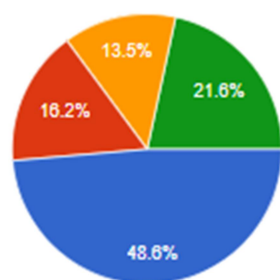
10 responses



- Looking up exercises myself and writing them down
- A dedicated app
- A dedicated website
- My friend gave it me

How do you track your workouts?

37 responses

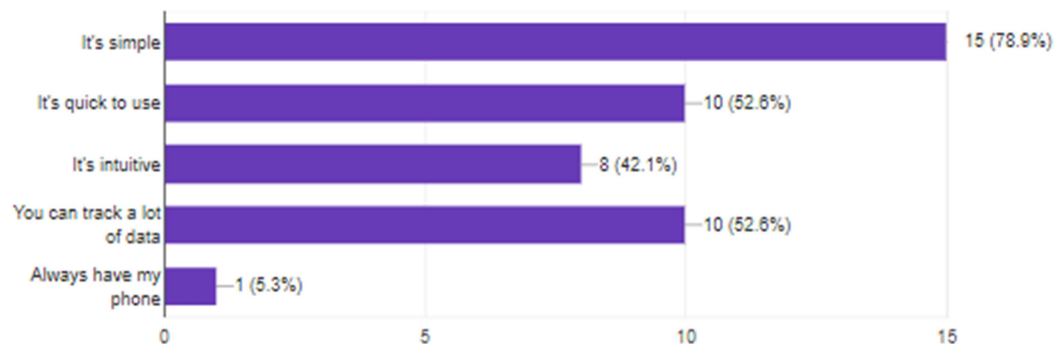


- I don't
- Pen and paper
- Note taking application
- Dedicated mobile app

Tracking your workout questions

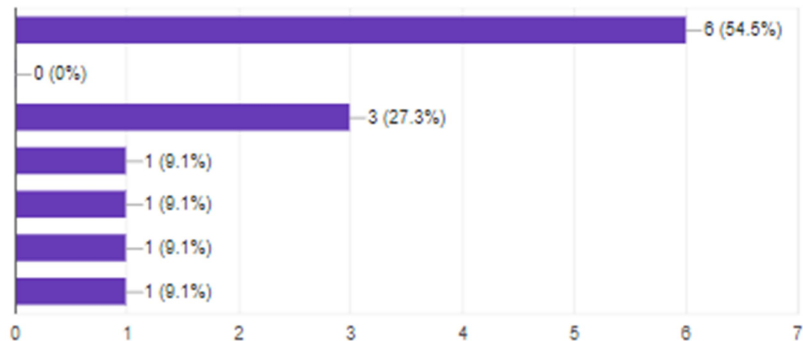
What do you like about this method of tracking workouts?

19 responses



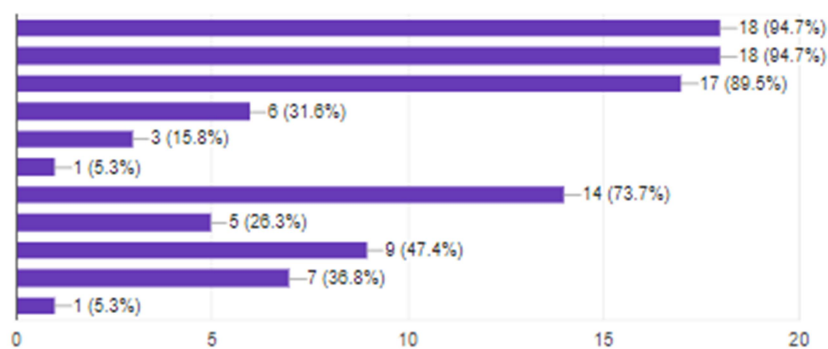
What do you dislike about this method of tracking workouts?

11 responses



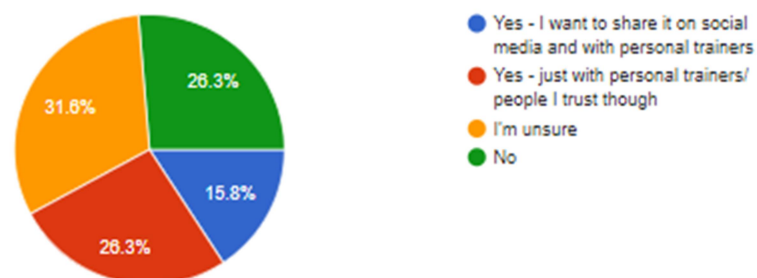
What data do you track? Please check all that apply.

19 responses



Would you like the option to be able to share the data you track?

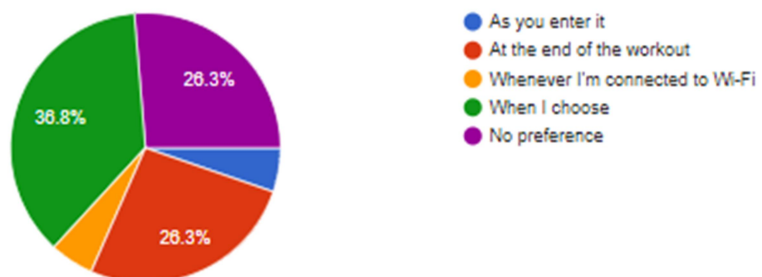
19 responses



When would you like to upload your fitness data?



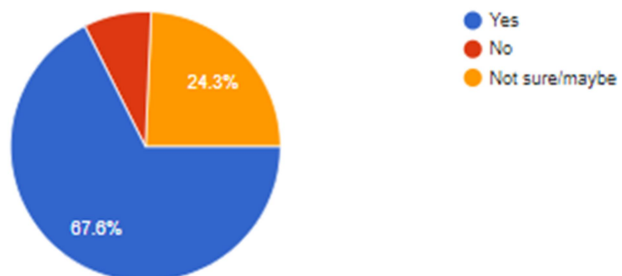
19 responses



Final question

Do you ever have to change what exercise you're meant to be doing due to lack of equipment e.t.c?

37 responses



Appendix C – User Evaluation Interview Guide

User Evaluation Interview Guide

Ask leading questions, allow the users to express their thoughts, and don't try to guide the conversation too much. Questions to ask to help probe for feedback:

Is there anything that was unclear to you?

Was the application easy to use? And was the navigation easy to understand?

What did you think about the exercise searching?

Do you feel like the application is missing anything you'd expect from a fitness application?

What are your thoughts on the website?

Is there any general changes or feedback you'd give?

Would consider using this application to track your own workouts?

Appendix D - User Evaluation Interview Answers

User Evaluation Interview answers

User 1	User 2	User 3	User 4	User 5
Unclear on what the % was for in the weights	Modify workout up button unclear	Nothing unclear, did take a minute to realise the % weight bit meant	Noticed difference in modify current workout and modify someone else workout screen	Application was easy enough to use
Unclear what unit weights were in	% is unclear what it means	Application was easy to use	Unsure what the modify workout up button was for until used	No units were confusing
Application was quick and the navigation made sense	Didn't understand advanced checkbox until they pressed it	Simple colours and text	No units given on personal measurements screen	No mention of if should record bar weight
Data Entry was also simple	Navigation was quick and easy the buttons made	Simple data entry	No weight units given	Likes how many measurements can be taken

	sense			
There was exercise suggestions for everything I could think of	Likes how the exercise timer expands when pressed	Likes expanding timer	Likes how you can see past measurements and workouts in text boxes	Likes how you can leave some of the logging fields blank
There was no calorie counting options	Likes how screens space is usually all used makes things easy to see and click	Likes how all data is backed up	Likes how things expand when in use	Likes the simplicity
Nothing specific to cardio	Likes the colour scheme on the app since it's simple	Likes how easy it is to see weights and reps done last time	Application is quick to load data like suggesting exercise	Likes how easy to press the buttons are for when in the gym
The website looks nice, liked the big graphs and customisation	Likes how workouts are shared	Likes the table for quickly seeing past workouts on the website	Colour scheme a bit bland but easy to read	Likes the website and the sharing but doesn't work well on their phone
The colours aren't very bright, although it makes everything easy to read	Likes how you save the graph data	Likes the charts on the websites but why limit to 3 exercises	Buttons are also easy to press	Would like a calculator for what weights to put on the bar
Wishes there were more charts on the website	Likes how the graphs looks	Would use the application themselves if it was available on iOS	Likes the charts	Would consider using this application after issues mentioned were fixed
Would use the application if their issues were addressed	Likes the customisation option on the website		Would consider using if units were sorted out	
	Would use the			

	application			
--	-------------	--	--	--

Appendix E – Project Specification

PROJECT SPECIFICATION - Project (SEGM) 2017/18

Student:	Sam Dawson
Date:	09/10/2017
Supervisor:	Mike Meredith
Degree Course:	Computer Science
Title of Project:	Analysis and tracking of Fitness data

Elaboration

My project is about giving people that are committed to fitness - particularly weight lifters an easy way to track a wide array of personal data and then to be able to easily view and analyse this data. I want to make it easy for people to be able to quickly track any level of detail they choose. This could just be basics like how much weight they lifted for however many reps and sets for a particular exercise. Or be as extreme as tracking time taken to complete a set, time they spent resting, total time for workout, the time their workout was at, the total volume on a certain body part, calories consumed in a day and body parts measurements along with their weight.

With a lot of these measurements it will allow a user to see their progress overtime over every part of their body in a very detailed manner. Such as the effects of volume/intensity on certain body parts and allow them to find the optimum amount of volume/weight for themselves.

Finding an easy to use application to track a lot of data very quickly is very hard - I've yet to find one and I just use an Excel spread sheet, as do many of my friends. I also want the app to be able to give you templates of successful/existing workout routines; these could be primarily user submitted and customisable once you've selected them – for instance if your gym is missing certain equipment you may want to choose a different related exercise. Additionally it could be possible to allow users to review workout programs that themselves or others have created and attach to the review how their own strength increased whilst performing the program. In future developments of my system if a lot of users were using this application it would be possible to extract some very interesting information such as how different amounts of volume and intensity of weight lifted at different frequencies affect muscle growth and strength increases.

Project Aims

- I aim to provide a sophisticated yet on the surface a very simple way to track a vast amount of data related to fitness.
- And to provide a way to analyse this data in great detail so you can find out what works best for any given user so they can reach their fitness goals.
- I want this information to optionally be sharable and for users to be able to choose what data they want to share, so you can share it on social media or with a personal trainer.

Project deliverable(s)

I will be delivering an Android application that will:

- Give a simple interface to track a large amount of fitness data
- Allow users to create and download other people's fitness plans (if shared)

A website that the users can login on that will:

- Pull the information that was collected on the Android device and then stored in a database
- Allow the user to view all this data in a user friendly way and give graphs/milestones

Action plan

Task	Deadline
Project Specification and Ethics Form	20/10/2017
Investigate similar products and the customer reviews corresponding to them	27/10/2017
Investigate different methods of questioning people and then act on one or more of these methods	3/11/2017
Investigate different technologies	6/11/2017
Create and send some data to a database	17/11/2017
Create a basic Android application that can be logged into using the database	24/11/2017
Send some basic data related to fitness to the database	1/12/2017
Create a basic website that you can login to	4/12/2017
Display in a graph some fitness data on the website	8/12/2017
Build up the basic applications to close to the finished product	31/1/2018
User testing and modifications	9/2/2018
Provisional contents page	9/2/2018
Draft Critical Evaluation	9/3/2018
Sections of a draft report	9/3/2018
Submit report body	11/4/2018
Submit project and report	12/4/2018
Demonstration	4/5/2018

Ethics

I don't think there are any potential ethics problems, the data is only sharable if the users allows for it to be. All data collected from any potential users for the purpose of helping shape how the product will look/feel will be anonymised. I will also give them detailed information on how their contributions will be used and if they give consent in light of this.

Appendix F – Ethics and Risks Checklist

Project (SEGM) [55-604708] Ethics and Risk Checklist

If the answer to any question is 'yes' the issue **MUST** be discussed with your project

Question	Yes/No
1. Does the project involve human participants? This includes surveys, questionnaires, observing behaviour, testing etc.	Yes
2. Does the project involve the use of live animals?	No
3. Does the project involve an external organisation? If yes, please write the name of the organisation here:	No
4. Does the project require access to any private or otherwise sensitive material?	Yes
5. Does the project require the reproduction (beyond normal academic quotations) of materials authored by a source other than yourself?	No

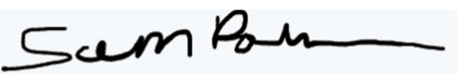
supervisor.

Ethics Checklist

Question	Yes/No
1. Does the project take any physical risks (such as electrical, lifting, travel)? If any risk is identified it must be discussed further with the project supervisor.	No

Risk Assessment

Adherence to SHU policy & procedures

Declaration
<p>I can confirm that:</p> <ul style="list-style-type: none">• I have read the Sheffield Hallam University Research Ethics Policy (available at http://www.shu.ac.uk/assets/pdf/research-ethics-policy.pdf)• I agree to abide by its principles. <p>Signature *  Print Name...Sam Dawson.....</p> <p>Date ...19/10/2017.....</p>

Appendix G – Stored Procedures with Testing

Stored Procedure Name	Description	Testing
Sp_Get1RM	Finds 1RM, for use in charts, was never implemented as a chart wasn't created to show this	Works as expected when results are compared to that of online 1RM calculates as it uses a standard accepted formula
sp_GetFavouriteExercises	Gets the favourite exercises set by a user	Tested by passing user token, results were the same as those stored against the user. When given an invalid token nothing is returned so works as expected
sp_GetPastPersonalGraph Data	Gets the past personal data updates of a user, given in a format specifically used to aid graphing the data.	Was passed a user token, the results were as expected, all data records are returned so this same function could be used to graph weight, height or anything else as desired
sp_GetPastWorkoutGraph Data	Uses the favourites that were set by the user to produce a set of data ready to be graphed (total volume of weight done of these exercises)	Data was returned as expected when passed a valid user token and could be successfully used to create a graph of the favourite exercises along with the dates they were recorded.
sp_GetPastWorkoutHistory TableData	Used to populate the table on the website that shows previous workouts	Data was returned as expected when passed valid user token
sp_GetPersonalData	Used to return latest records of each piece of personal data for the application to use as history, doesn't return date of each piece of data	Data was returned as expected when passed a valid user token
sp_GetWorkoutDayData	Returns the workout template for the current day the user is set to	Data returned without historic records as expected when called with a valid user token
sp_GetWorkoutDayNames	Used to get day names in a workout along with the day number for use in the settings page so a user can	Names returned with day number as expected when passed user token

	change their current day	
sp_GetWorkoutDayWithHistory	Returns the workout template with the history of the last time the workout was completed, or if not available then the initial guideline data for the day the user is set to	Data returned with historic records, or initial set data as expected when called with a valid user token
sp_GetWorkoutTemplate	Returns a workout template when one is selected on the search screen	When passed a valid workout template PK and a user token a workout template is returned as expected
sp_GetWorkoutTemplates	Returns a list of suggestions of workout templates with some minor details about the template	When provided a search parameter user workout templates are returned
sp_LoginAndCreateToken	Receives E-mail and password if valid creates a user token and stores it then returns this user token	When provided with a valid E-mail and password a user token is either returned if a valid one already exists (created in the last two hours) or if not it's created and then returned
sp_LookUpExercise	Searches for exercise suggestions from a supplied name	Returns suggestions as expected when provided with a name
sp_PostPersonalData	Takes a number of optional parameters and then inserts a record for the given user with the current date time	When supplied with a token and any variety of measurements all are successfully stored in the database.
sp_RegisterUser	Registers a user by creating a record for the user from a supplied E-mail, password and name	When passed the required parameters the record is created as expected and the password is stored as a SHA2-512 hash as expected.
sp_SetUserExercises	Sets the favourite exercises for a user 1-3 exercises can be provided	When passed 1, 2 or 3 exercisePKs along with a user token the user favourite exercises are updated as expected

sp_SetUserWorkoutDay	Used for setting a user's current workout day manually	When provided with a token and a day number the day is set to the current day number as expected
sp_UserNextWorkoutDay	Increments the current day the user is on, called after a workout is completed. If there is no day up next it looks back to the start	When provided with a user token the day number is incremented as expected
sp_ValidateToken	Checks if a token is valid, can be used for authentication	When provided with a token it correctly identifies if the token is valid or not