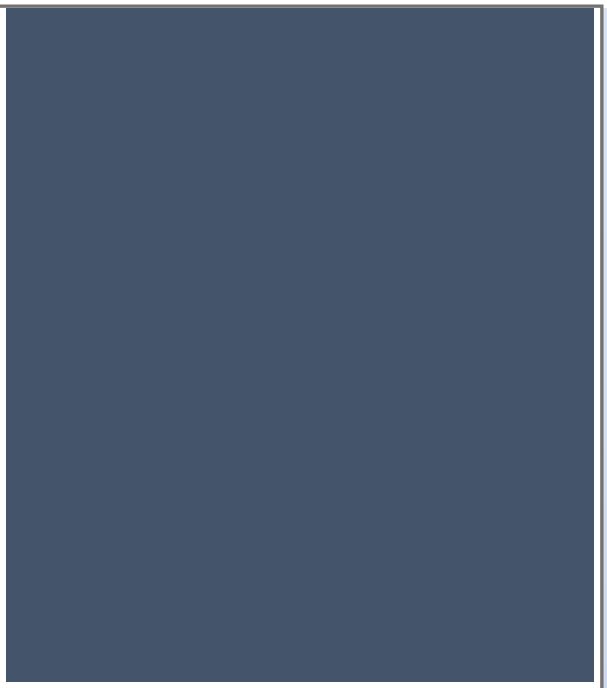


Exercise Web Companion

Software Development Project

Paul Holt

Contents

Version	***Adjustments***	***Date***
1.0	• Creation of Document • Introduction	12/01/21
2.0	• Creation of Glossary • Writing of requirements section • Adding Appendix A, B, C • Added Risk Section	02/02/21
3.0	• Added MoSCoW Diagram • Justification For MoSCoW Diagram.	12/02/21
4.0	• Conclusions to primary research	17/02/21
5.0	• Metholology section completed • Moved user stories and use cases.	18/02/21
6.0	• Implementation section for create an account	23/02/21
7.0	• Review section for create an account	06/03/21
8.0	• Implementation and review section for plan a workout	03/04/21
9.0	• Changes to all use case diagrams to show previous sections. • Implementation and Review sections for Scheduling a workout • Updated Methodolgy to have quality asurance, expicitly labelled functional and non functional requirements. • Made reference to github and config management • Moved Glossary	04/04/21
10.0	• Added Tools used • Updated implementation section to have; additonal tools, requirements filled, problems and breakdown • User Reviews • Quality Assurance • Added content Aggretation chapter	05/04/21
11.0	• Conlusion added • Moved Breakdown sections to appendix • Proof read – minor changes	07/04/21
12.0	• Moved User Stories To Appendix	12/03/21
13.0	• Reworked Methodology Section • Added Introduction Section	19/04/21
13.1	• Reworked Introduction Section	20/04/21

1.Glossary

WORD	DEFINITION
AGILE FRAMEWORK	Agile methodology is a project management framework, used by teams to iteratively and incrementally complete tasks and projects. (Target Internet, 2021).
CONTENT AGGREGATORS	A content aggregator is an entity that pulls together web content, applications, or both from online sources for reuse or resale.
ENTITY RELATIONSHIP MODEL	Entity Relationship Model (ER Modelling) is a graphical approach to database design. It is a high-level data model that defines data elements and their relationship.
IFRAME	An iFrame is an inline frame used inside a webpage to load another HTML document inside it.
ISO/IEC	International Organisation for Standardisation/International Electrotechnical Commission
JQUERY	jQuery is a lightweight, "write less, do more", JavaScript library. The purpose of jQuery is to make it much easier to use JavaScript on your website.
MOSCOW METHOD	The MoSCoW method is a prioritization technique used in management, business analysis, project management, and software development to reach an understanding on the importance of each requirement.
PB	Personal Best.
QUALTRICS	Qualtrics is a web-based software that allows the user to create surveys and generate reports.
SCRUM	Scrum is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.
SEQUENCE DIAGRAM	A sequence diagram is a type of interaction diagram which describes how—and in what order—a group of objects work together.
UML DIAGRAM	A UML diagram is a diagram based on the UML (Unified Modelling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand a system.
USE CASE DIAGRAM	The purpose of a use case diagram in UML is to demonstrate the different ways that a user might interact with a system.
USER STORIES	A user story is an informal, general explanation of a software feature written from the perspective of the end user. Its purpose is to articulate how a software feature will provide value to the customer (Rehkopf, 2021).

2.Introduction

The goal of the project is to create a functioning prototype web application to aid in the process of creating, managing and scheduling workout plans. This is to help both experienced and amateur people in the field of exercise to prompt a better and healthier lifestyle. Research conducted within the project will help to uncover what motivates people in this regard and will then be used to inform a set of requirements to achieve the aforementioned goal.

A core aim of this project then, must be to create a one stop shop, for beginners to start their journey into a healthier lifestyle. Combining the core features represented in the best of the business and adding new functionality such as providing user specific YouTube videos for tips and guidance.

This report contains a description of the methodology used to manage and conduct the project, a detail description of the requirements and where they were derived from, how the requirements were implemented and a review of how well the requirements were met based on the assigned fit criterion and user feedback. All code will be incrementally uploaded to the GitHub Repository linked below.

Link to Project Repository: https://github.com/pjholt1234/ExerciseWebCompanion

3.Background

Obesity is on the rise within the west, around three quarters of people ages 45-74 in England are overweight or obese (Baker, 2021). 2018 - 2019 saw a rise of 4% in hospital admissions directly attributable to obesity, while 67% of adults and only 47% of young people were considered active as per government guidelines (NHS, 2019). With such strong ties to reducing life expectancy, the obesity problem in the west could be considered an epidemic, meaning the importance of leading a healthy life is more prevalent now more than ever. Exercise is a corner stone of this and yet for so many, can be a difficult and confusing task. This is reflected by the lack of people engaging with exercise in the UK, “37% of people in the UK never exercise or play sport” (Cooke, 2018).

The exercise application world is well populated, with brands such as Strava and Nike at the forefront, Strava currently has 55 million users and reportedly adds one million every month (CURRY, 2020). “During a March 2018 survey, 42 percent of U.S. adults stated that they used some sort of digital technology to measure fitness and health improvement goals” (Statista Research Department, 2018).

4.Methodology

4.1.Frameworks Used

During the project lifecycle I will be making commits to a GitHub repository, this will act as version control for the application. The report will document when any changes are made in the version control section, these two processes will act as my configuration management.

Initial research is to be conducted regarding the other competitors within the field of "exercise applications". This research will help to guide the scope of the project as it will educate me on the sort of features which the application should contain. Once this is completed a plan will be drawn for the primary research phase, the questions will help to build out the scope and better understand the requirements of the project. Questions will revolve around how people chose to plan their exercise and what motivates them.

The project will follow an "agile" framework, this is to iteratively complete tasks as the project develops. In particular, I will be using the methodology "SCRUM", this method allows for flexibility this is important as the time scale is restrictive. SCRUM projects are organised into sprints, each "sprint" focuses on a feature or in this case "module" within the project. A sprint involves requirements, design, implementation, and review phases. To compliment SCRUM, I will be applying "Decision Science Research Methodology" (DSRM), to create the report. This framework allows the report to develop from iteration to iteration in conjunction with the project.

4.2.Report Structure

The structure of the report will be as follows:

- ***Background***.
- ***Methodology***.
- ***Scoping Requirements*** – This is where research will be contained alongside developing and prioritizing the requirements.
- ***Tools and Services Used***.
- ***Risks.***
- ***Project iterations –*** Containing, requirements, design, implementation, and review phases (see section 4.3 for more)
- ***Quality Assurance –*** Testing non function requirements.
- ***Conclusion.***

4.3.Iteration Structure

Requirements Phase

Within this project, the functional requirements are captured in the research section processed into "user stories". They will be labelled "name of the requirement" + "FR" in the user stories, to ensure continuity throughout the requirements and review sections. These features will be derived from problems or conditions voiced within the research section. User stories provide an acceptance criterion which will be used within the review process. Bridging the gap between the requirements and the design phase is the "Use case diagram". The aforementioned user stories are taken and used as functions within the use case diagram.

Design Phase

This phase consists of creating a "sequence diagram", this sequence diagram helps to show the relationship between objects aiding the implementation process. Other models used within in this section could be "entity relationship model" used for designing the database and a "UML diagram" to show the relationship of classes across the whole system.

Implementation Phase

Implementation consists of creating the application, this could involve programming or database creation alongside a description of how it was implemented, this is then separated into 3 sections; the "Additional Tools Used", the "Requirements Filled" where functional requirements are paired with the corresponding functionality, and finally the breakdown where the functionality is explained.

Review Phase

The review phase is spilt up into 2 sections the first being verification, this consists of testing if the program meets the requirements by completing the acceptance criterion set in the user story section for the FR's or NFR's. The second is user review where the application will be tested by users. If the application fails in any section, the process starts again from the design phase.

5.Scoping Requirements

5.1.Research

5.1.1.Exploring Competitors - Secondary Research

See Appendix A for Research.

At the beginning of the project some initial research was undertook to better understand the market of exercise applications. 4 of the most popular exercise applications were selected from the Android Play and Apple App store, Map My Run, Fitness Buddy, Daily Strength, Nike Training Club. The purpose of this research was to grasp the scope of the project. Some features that were of particular interest were, the ability to set personal goals, track stats, create custom workouts and visualisation of achievements and goals. Poor features included, features not being fully developed or behind a pay wall and content not being accessible to novice users due to complexity **A.1.** These features will form the basis of the primary research to further uncover the requirements of the application.

5.1.2.Exploring Competitors – Conclusions

Some key takeaways from this research were that the application must be accessible for all levels of knowledge, additionally the design must be simple enough to be intuitive. Most exercise apps allow for users to create and track goals, this seems to be a core component and should be a requirement of my application. Some Applications act as "content aggregators", this is an interesting feature and should be explored.

5.1.3.Survey - Primary Research

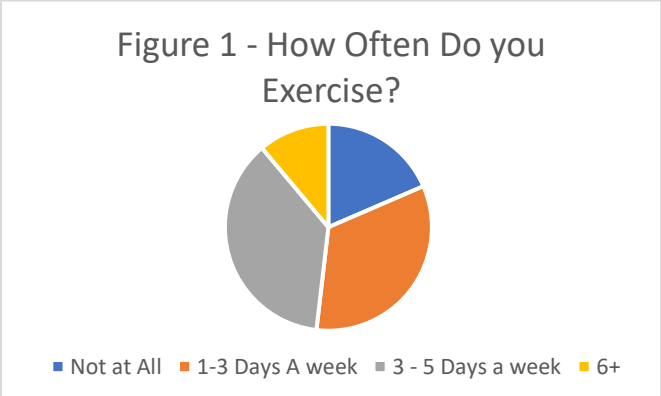
See Appendix B.1 for Research Plan.

See Appendix B.2 for Research Data.

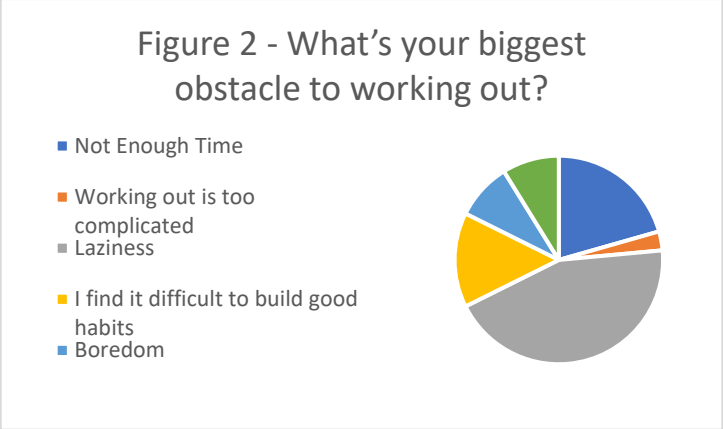
The Research was conducted on "Qualtrics", this was to ensure that data was stored securely, and participants remained anonymous. The research to be undertaken was deemed to be low risk and the ethics documentation can be found in appendix C.1. The purpose of the primary research was to further understand the requirements of a broad target audience for the exercise companion, the participants ages ranged from 18 to 60+ and coming from a range of backgrounds in regard to their knowledge and experience with exercise. This was important to ensure that requirements were captured for the majority of possible users however, primarily I wanted to uncover the motivating factors for a novice or an amateur to exercise. The questions were derived from both the secondary research and to aid the process within my methodology.

5.1.4.Survey – Conclusions

Question 1 established that of the participants 14 do not exercise frequently enough B.2.1 (**see figure 1**). Reasons and obstacles causing participants to not workout, included not enough time and difficulty building good habits however, the largest share was down to laziness B.2.3. (**see figure 2**). The web companion could help prevent this by providing useful



methods to track progress and set goals. This is reinforced by the results of B.2.6, where the majority of participants said that the aforementioned feature would be useful. This was also a highlighted feature from competitors applications meaning it should be a focal point of this application.



Other methods in which the application could tackle laziness is by linking with 3rd party calendar services, section B.2.10 highlights that within this survey group 16/33 participants use google calendar regularly, participants expressed some interest in automated reminders B.2.5. An additional obstacle to working out was lack of knowledge, 9 participants expressed difficulties in understanding how to effectively exercise. An example of an obstacle was as follows, "beginning weightlifting in the gym can be difficult if you don't have anyone that is able to tell you the best exercises to do for certain muscles". It is clear then that the web companion must suggest workouts and useful tips for beginners to ensure that amateurs to exercising are supported. When asked what method is used to find information out regarding exercise, the majority of participants said that they use Google or YouTube. (**See Figure 3**), meaning that this could be integrated in some way streamlining the learning process and removing another obstacle to exercise. Please see section 4.2.2 for feature prioritization where it has reference to the findings of section B.2.9. of the appendix.

Figure 3	Totals
YouTube	18
Facebook	0
Google	16
Friends or Family	9
Books	2
Other	4

5.2.Functional Requirements

5.2.1.Features and Justification

Log in Securely (FR 1)

- Being able to login to a profile is key to the whole project, it ensures the users information is able to be stored and retrieved.

Create an account (FR 2)

- Allowing users to be able to create a profile is equally as important as it allows the system to grow.

Add Workouts (FR 3)

- Adding Workouts is the centre of the whole project, users must be able to select and add exercise to their custom plans, this follows in the footsteps of other competitors and is reflected in the primary research as a desired feature.

Sort Workouts (FR 4)

- When users are creating workouts, they should be able to sort and change the order, this allows for them to create custom plans which is a cornerstone of the whole project, without this feature users would be stuck with premade plans which would put the project at a disadvantage in comparison to its competitors.

Save Workout Plans (FR 5)

- Users must be able to save workout plans, see FR 3 + 4 for justification.

Retrieve and Edit Workout Plans (FR 6)

- Once saved users must be able to retrieve their plans so that they can make changes, see FR3 + 4 for justification.

Schedule Workout Plans (FR 7)

- Laziness was one of the primary factors which led people to not exercising, when asked what would encourage them building good habits and encouragement were two responses, this problem could be satisfied by allowing users to schedule plans.

Delete Workout Plans (FR 8)

- Since users are creating and scheduling plans, they need to be able to remove them.

Alter Workout Plans (FR 9)

- Similar to FR 8, users need to be able to adjust their plans if they aren't at a desirable time.

Logging PB's (FR 10)

- When asked what the most desirable features for this application, the highest scoring feature was logging personal bests and tracking weight. *See appendix B.2.9*

Seeing Exercise breakdown (FR 11)

- Similar to FR 10, users like statistics and find it a strong motivator.

Weight tracker (FR 12)

- See FR 10.

Visual Aid's (FR 13)

- Over 2/3's of users when asked whether or not visual aids to stats would be useful and desirable replied either moderately useful or extremely useful. *See appendix B.2.6*

Tailored personalised tips (FR 14)

- Participants expressed difficulties in understanding how to effectively exercise, when asked how they would figure these problems out the majority said they would consult YouTube or Google. The application could embed tailored YouTube content.

Add Reminders (FR 15)

- Many participants said that reminders would be useful in encouraging exercise and since almost 50% of participants use google calendar, the website could integrate with it to provide useful reminders.

Log Workouts (FR 16)

- This function would facilitate FR 15.

Plan a run tool (FR 17)

- The plan a run tool could enable users to discover and create running routes.

5.2.2.Grouping Features

Figure 4 groups features into "modules". An example of a module is the "Plan a workout" section of the web companion, this consists of features and functions which coexists to allow for the user to; plan a custom workout, add new workouts, and save them for later. The web companion is composed of 7 key modules; "Creating a Profile", "Plan a workout", "Scheduling a Workout", "Generating statistics", "Content Aggregation" and possibly "Plan a run" and "Google Calendar Integration". The modules will be organised using MoSCoW, this will help to inform the flow of the project as it ranks the desirability of features.

Figure 4 – Module Feature Table

Creating a Profile	**Plan a workout**	**Scheduling a Workout**	**Generating statistics**	**Content Aggregation**	**Google Calendar Integration**	**Plan a run**
Log in Securely **(FR 1)**	Add Workouts **(FR 3)**	Schedule Workout Plans **(FR 7)**	Logging PB's **(FR 10)**	Tailed personalised tips **(FR 14)**	Add Reminders **(FR 15)**	Plan A Run Tool **(FR 17)**
Create an account **(FR 2)**	Sort Workouts **(FR 4)**	Delete Workout Plans **(FR 8)**	Seeing Exercise breakdown **(FR 11)**		Log Workouts **(FR 16)**	
	Save Workout Plans **(FR 5)**	Alter Workout Plans **(FR 9)**	Weight tracker **(FR 12)**			
	Retrieve and Edit Workout Plans **(FR 6)**		Visual Aid's **(FR 13)**			

With this in mind, the rest of the prioritization is based off the data gathered from the primary research. Section B.2.9 of the Appendix covers the results of the question where participants were asked to rank possible features. An example of where dependencies come into play when ranking priority is all features involving creating a profile. This feature was not mentioned with in the research question, however, is one most important feature as all other modules requires it to exist. Plan A run was placed into the "Could Have" section as it would require a sizable amount of time to complete and would put the project at risk of not being completed in time if placed higher.

5.2.3.MoSCoW

Must Haves:

- Log in Securely **(FR 1)**
- Create an account **(FR 2)**
- Add Workouts **(FR 3)**
- Sort Workouts **(FR 4)**
- Save Workout Plans **(FR 5)**
- Retrieve and Edit Workout Plans **(FR 6)**
- Schedule Workout Plans **(FR 7)**
- Delete Workout Plans **(FR 8)**
- Alter Workout Plans **(FR 9)**
- Logging PB's **(FR 10)**
- Seeing Exercise breakdown **(FR 11)**
- Weight tracker **(FR 12)**
- Visual Aid's **(FR 13)**

Should Have:

- Tailed personalised tips **(FR 14)**.

Could Have:

- Add Reminders (using google calendar) **(FR 15)**
- Log Workouts (using google calendar) (**FR 16)**
- Plan A Run Tool **(FR 17)**

Will Not Have:

- Dietary help

5.3.Non-Functional Requirements

Non-functional requirements will be generated to use as quality assurance, these will be designed around the ISO/IEC 9126 (British Standards Institution, 2001) and will be contained under the requirements heading. A non-functional requirement will be generated for the following 6 categories, functionality, reliability, usability, efficiency, maintainability, and portability. These features will be reviewed at the end of the project under the "quality assurance" chapter. The format of the requirement is as follows; "NFR + number" + "Name of requirement" + "test case".

Functionality – This defines an attribute of a set of functions within a system.

- *NFR1 Security* – All passwords must be kept private and hashed to prevent a data breach.

Reliability – This defines attributes that measure the software's ability to maintain its level of performance.

- *NFR2 Reliability* – The website must have the same loading speed and performance over the course of 1 week.

Usability - A set of attributes that assess the level of effort require for use.

- *NFR3 Usability* – The user must be able to intuitively create a new "chest plan" with 5 workouts, save it, and then schedule it for today within 5 minutes.

Efficiency – A set of attributes that dictates the speed of an application.

- *NFR4 Efficiency* – Each page should load within 5 seconds.

Maintainability – A set of attributes which defines how easy the application is to repair or change.

- *NFR5 Rebuild* – The application must be able to be rebuilt on a new server within 10 minutes of acquiring the files.

Portability – A set of attributes which explains how the application could be moved from one area to another.

- *NFR6 Compatibility* – The website must be able to perform on 3 different types of hardware, a desktop, laptop, and smartphone.

6. Risks

There are a number of risks which the project faces at this time below they are listed with their contingency.

Risk 1 – Failure to complete the project within the time requirements.
Contingency 1 – Ensure that the project is well managed and meets milestones, also ensure that the project has room to grow without threating its outcome.

Risk 2 – Failure to fully understand the tools and services required to complete the project.
Contingency 2 – Ensure that any choice of application or service is well researched and planned.

Risk 3 – Limitations of access to support due to the pandemic.
Contingency 3 – Keep in regular contact with supervisor and ensure communication is maintained within a schedule.

Risk 4 – Limitations of access to possible test users due to the pandemic.
Contingency 4 – Ensure the website is either hosted somewhere where individuals can access it or perform testing over a zoom/teams meeting.

7. Tools and Services Used

These are the tools that are applied in all of the sprints, if any different tools are used, it will be mention at the beginning of the necessary implementation sections.

Sublime Text 3 - A lightweight, cross-platform code editor known for its speed, ease of use, and strong community support (Sublime Text, 2021). This is used as the text editor, and where the majority of the programming takes place.

PHPMyAdmin - PhpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web (phpMyAdmin, 2021). This is how I interface with my database.

XAMPP - A free and open-source cross-platform web server solution stack package developed by Apache Friends (Apache Friends, 2021). XAMPP Is what I use to host the website locally during development.

HTML - HTML stands for Hyper Text Markup Language. HTML is the standard markup language for creating Web pages.

CSS - CSS stands for Cascading Style Sheets. CSS describes how HTML elements are to be displayed. This is used for frontend styling.

JavaScript - JavaScript can update and change both HTML and CSS, this is the language used to manage frontend DOM objects and some computing.

jQuery - A fast, small, and feature-rich JavaScript library. It makes HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API. I used this to make manipulating front end objects easy and adding additional functionality such as interactive UI elements.

Ajax - With Ajax, web applications can send and retrieve data from a server asynchronously (in the background) without interfering with the display and behaviour of the existing page. I used this in some cases where the JavaScript had to interact with the database directly without submitting a php form.

PHP – A widely-used open-source general-purpose scripting language. This is used this for the backend of the website to interface with the MySQL database.

MySQL - An open-source relational database management system. This was used to develop the database which is the backbone of the application.

QSEE Modelling Tool – This was used to develop entity relationship models of the database. This helped to show the relationships between tables.

LucidChart – Lucid chart is an online tool for making various technical diagrams, I used to make use case diagrams, UML diagrams, and sequence diagrams (Lucidchart, 2021).

8.Creating A Profile

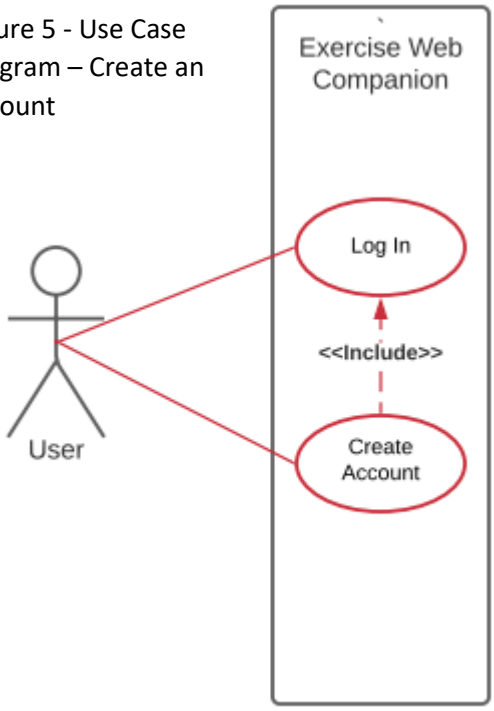
8.1.Requirements

8.1.1.User Stories

See Appendix G.1 for User Stories.

8.1.2.Use Case Diagram

Figure 5 - Use Case Diagram – Create an account



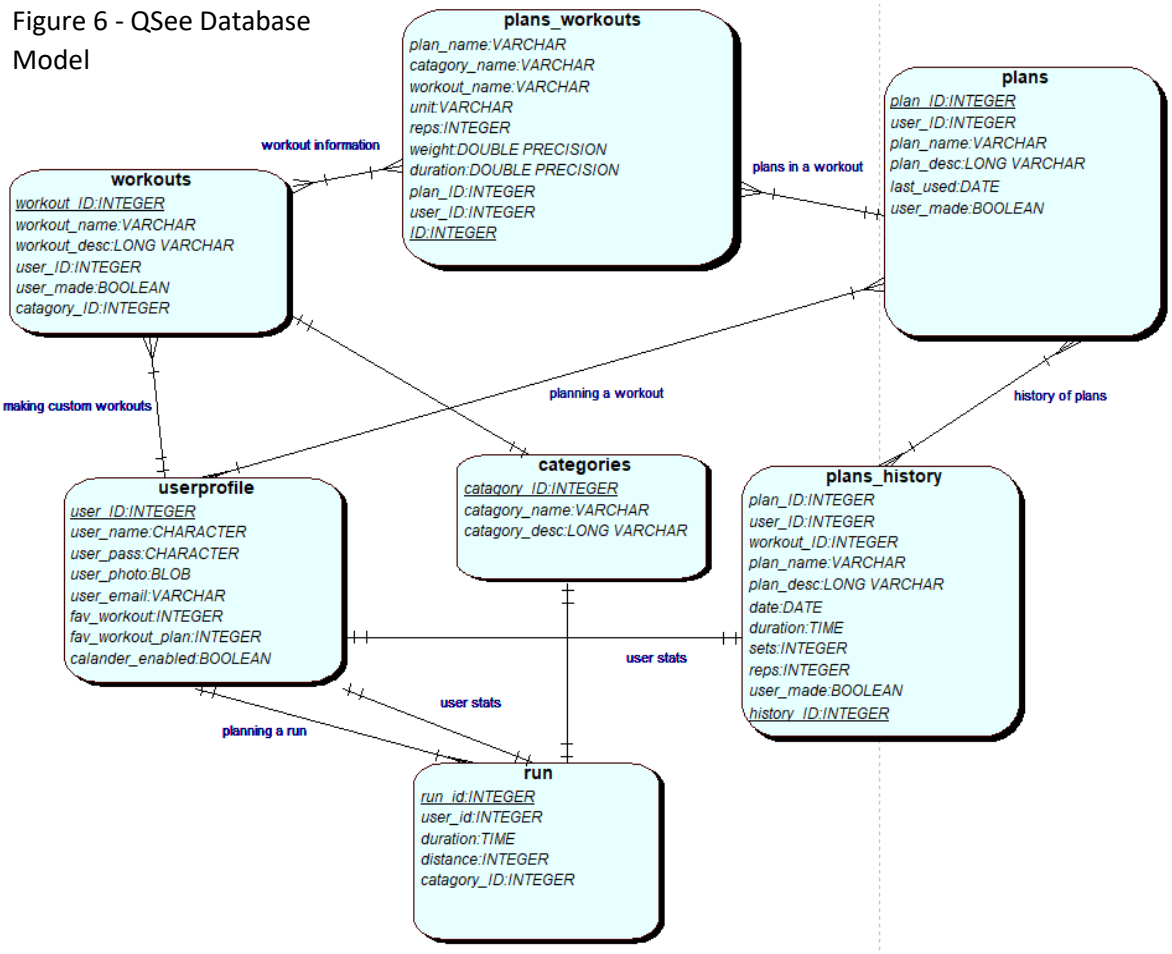
Key

Grey Outline = Previous Sprint

Red Outline = Current Sprint

8.2.Design

Figure 6 - QSee Database Model



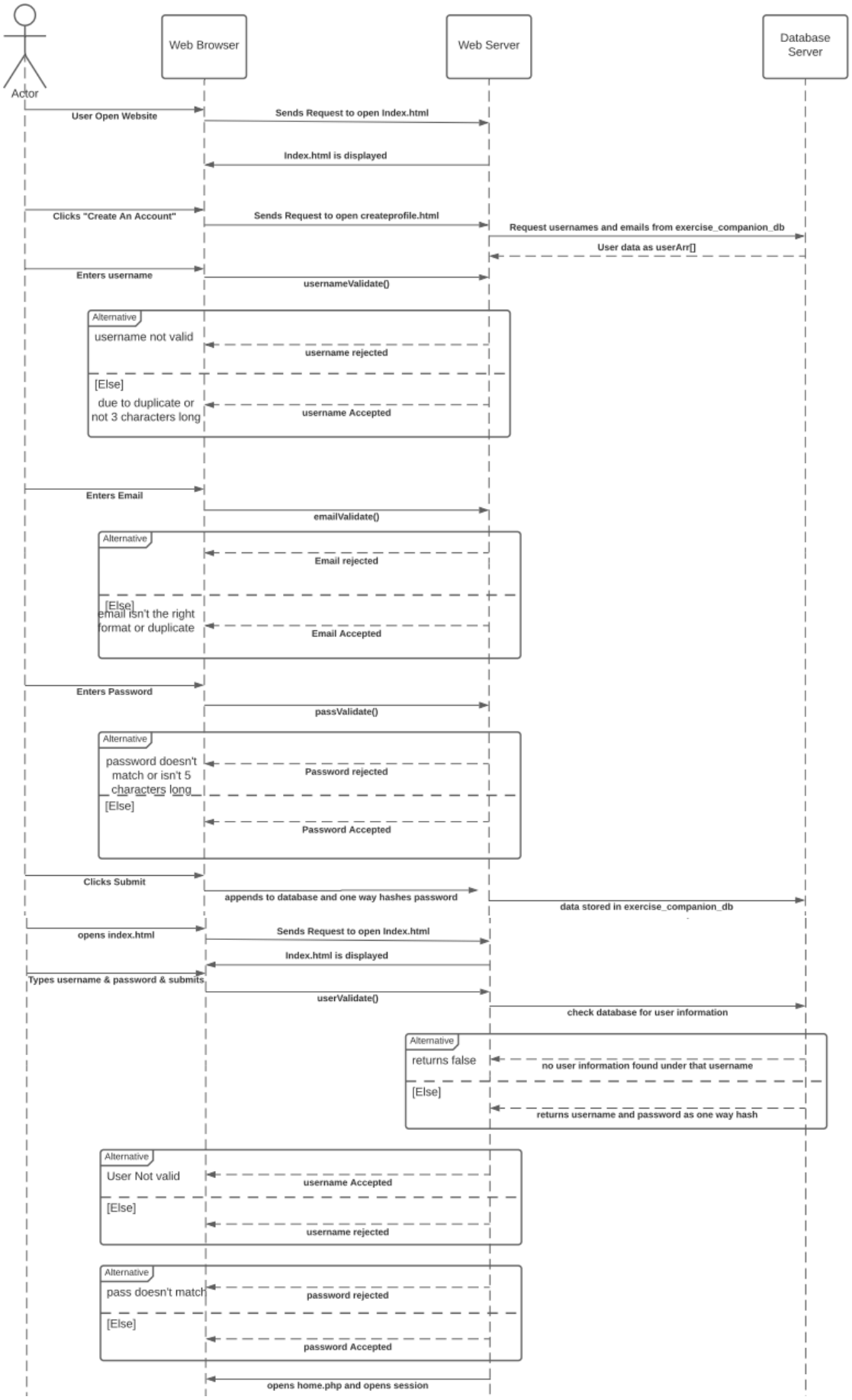


Figure 7 – Sequence Diagram – Create A Profile

8.3.Implementation

8.3.1.Additional Tools used

- N/A

8.3.2.Requirements filled.

Figure 7	
Requirement	Functionality
Log in Securely (FR 1)	The login page as a 1-way hash preventing anyone from seeing a user's password.
Create an Account (FR 2)	In the index.html page you are able to create an account as long as you have a valid email address, username and password.

8.3.3.Problems and lessons learnt.

1. Before creating this page, I didn't know how to protect passwords, I learnt how to use php's password_hash method.

8.3.4.Breakdown.

See Appendix D.1 for Create an Account implementation breakdown.

8.4.Review

8.4.1.Verification

See Appendix E.1 for Verification Table.

8.4.2.User Review

See Appendix for F.1 User Reviews Table.

9.Plan a Workout.

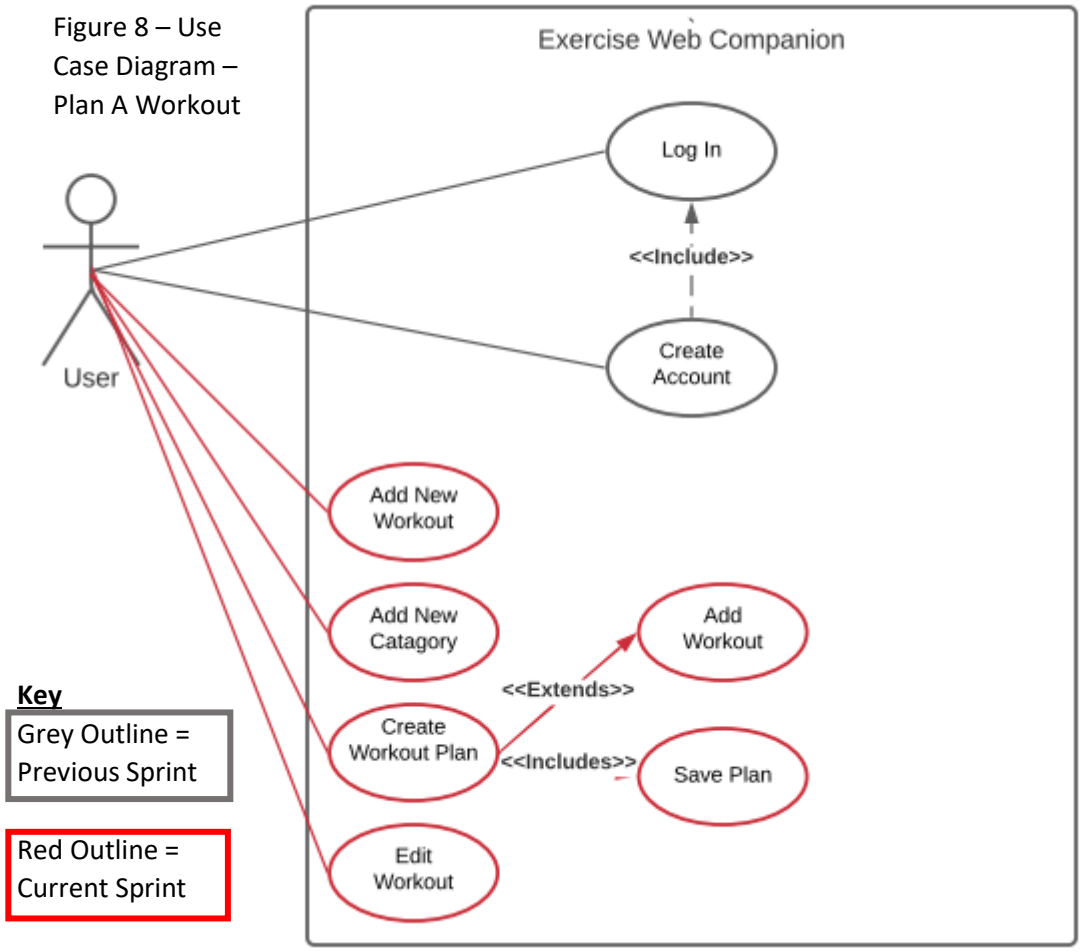
9.1.Requirements

9.1.1.User Stories

See Appendix G.2 for User Stories.

9.1.2.Use Case Diagram

Figure 8 – Use Case Diagram – Plan A Workout



9.2.Design

Figure 10 - QSee Database Model Plan A workout

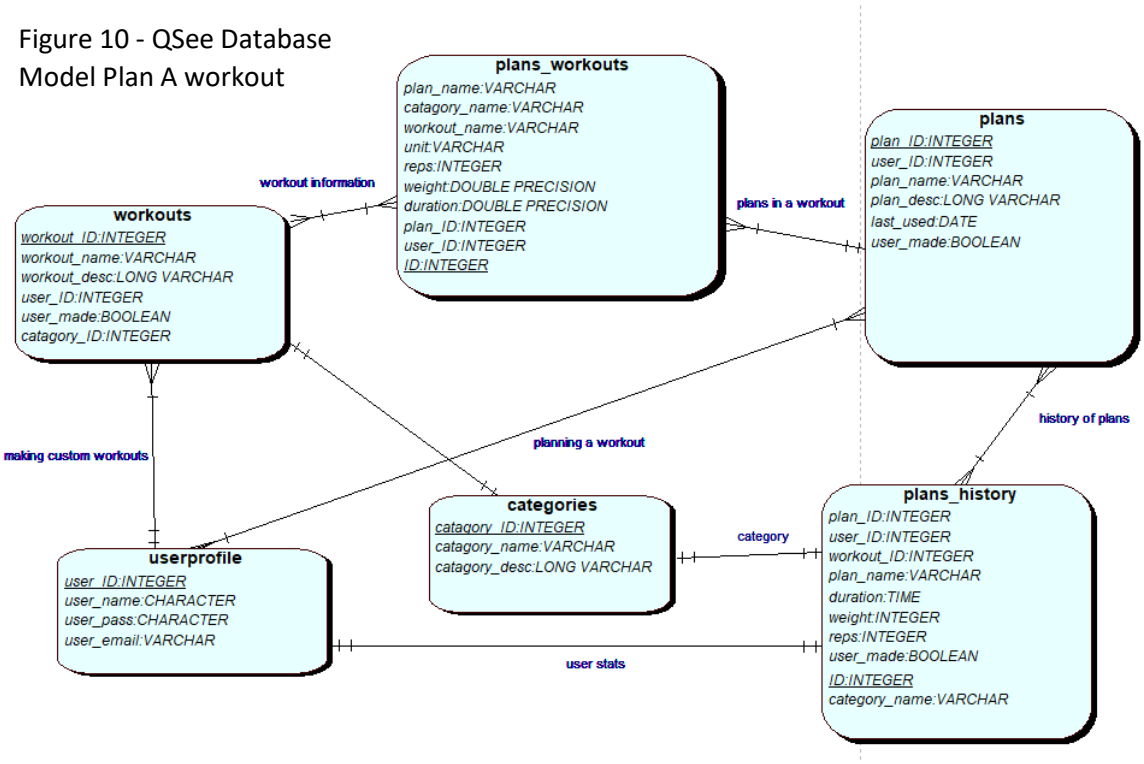
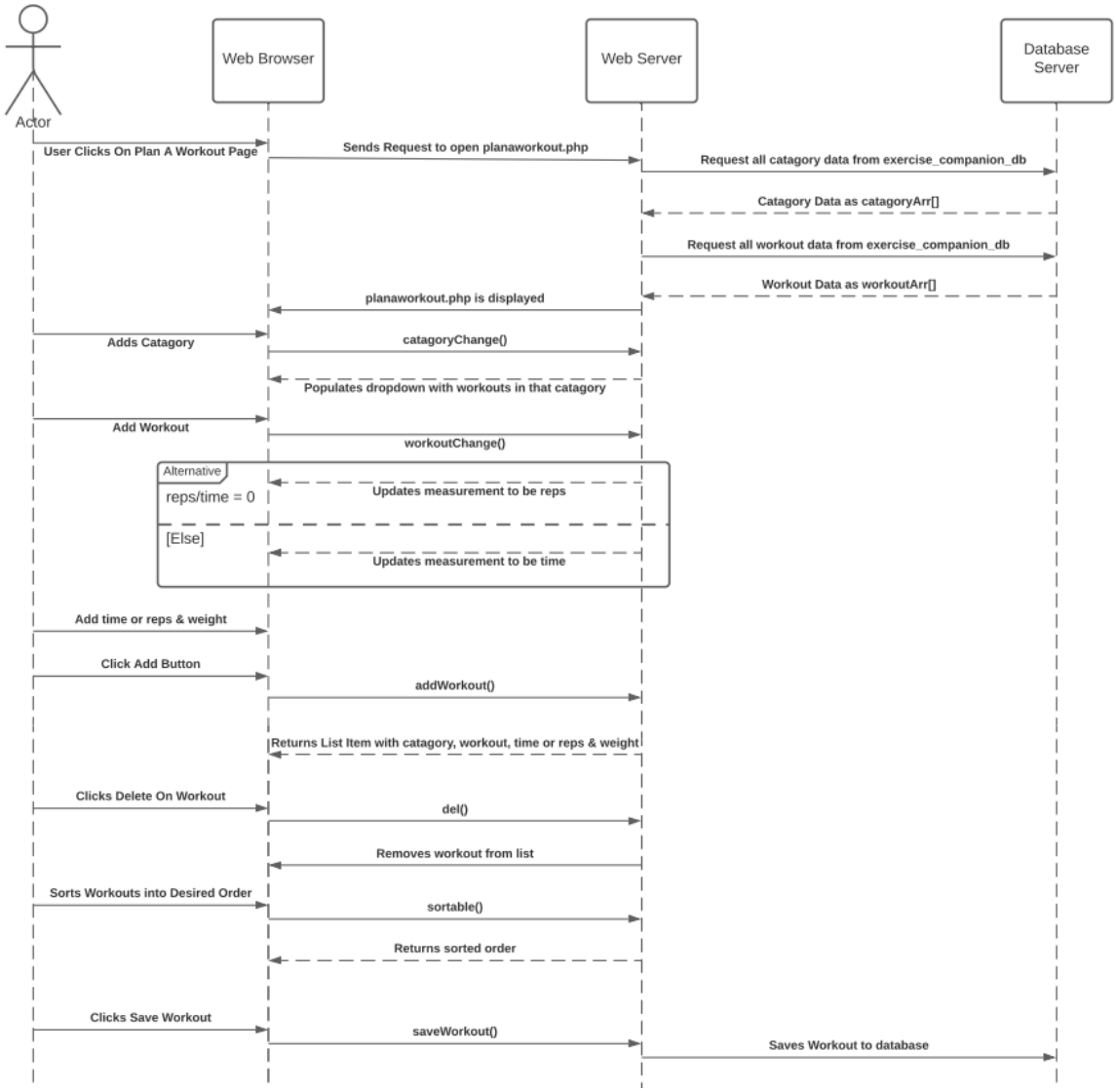


Figure 9 - Sequence Diagram – Plan A Workout



10.Implementation

10.1.1.Additional Tools used

- N/A

10.1.2.Requirements filled.

Figure 11	
Requirement	Functionality
Add Workouts (FR 3)	When the user selects custom from the drop down the pop up appears allowing them to enter new workouts and categories.
Sort Workouts (FR 4)	The jQuery sortable feature allows users to easily organise and sort their workouts.
Save Workout Plans (FR 5)	The save button allows users to save their workout plans.
Retrieve and Edit Workout Plans (FR 6)	The jQuery accordion lets users edit and delete workout plans

10.1.3.Problems and lessons learnt.

1. Since I'd never used jQuery before I had to completely learn it from scratch, which was difficult, however this knowledge will be useful throughout.
2. Working out how to dynamically assign elements IDs was difficult but, I came up with a solution to concatenate a count variable with the type of element in question, meaning I could reference it. This was a breakthrough moment for the project however simple it may seem.

10.1.4.Breakdown

See Appendix D.2 for Plan a Workout implementation breakdown.

10.2.Review

10.2.1.Verification

See Appendix E.2 for Verification Table.

10.2.2.User Review

See Appendix for F.2 User Reviews Table.

11.Scheduling a Workout

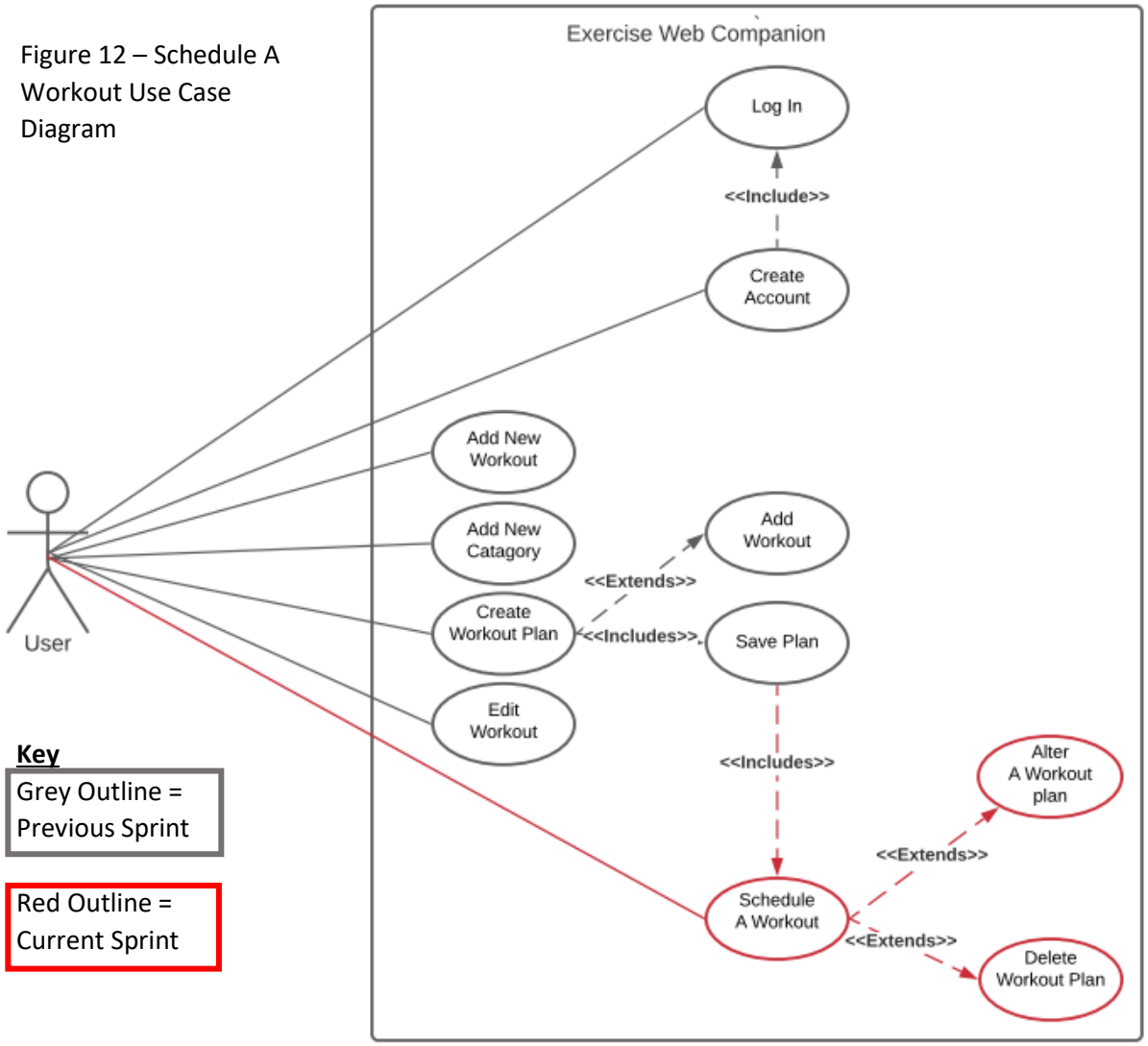
11.1.Requirements

11.1.1.User Stories

See Appendix G.3 for User Stories.

11.1.2.Use Case Diagram

Figure 12 – Schedule A Workout Use Case Diagram



11.2.Design

Figure 13 - QSee Database Model

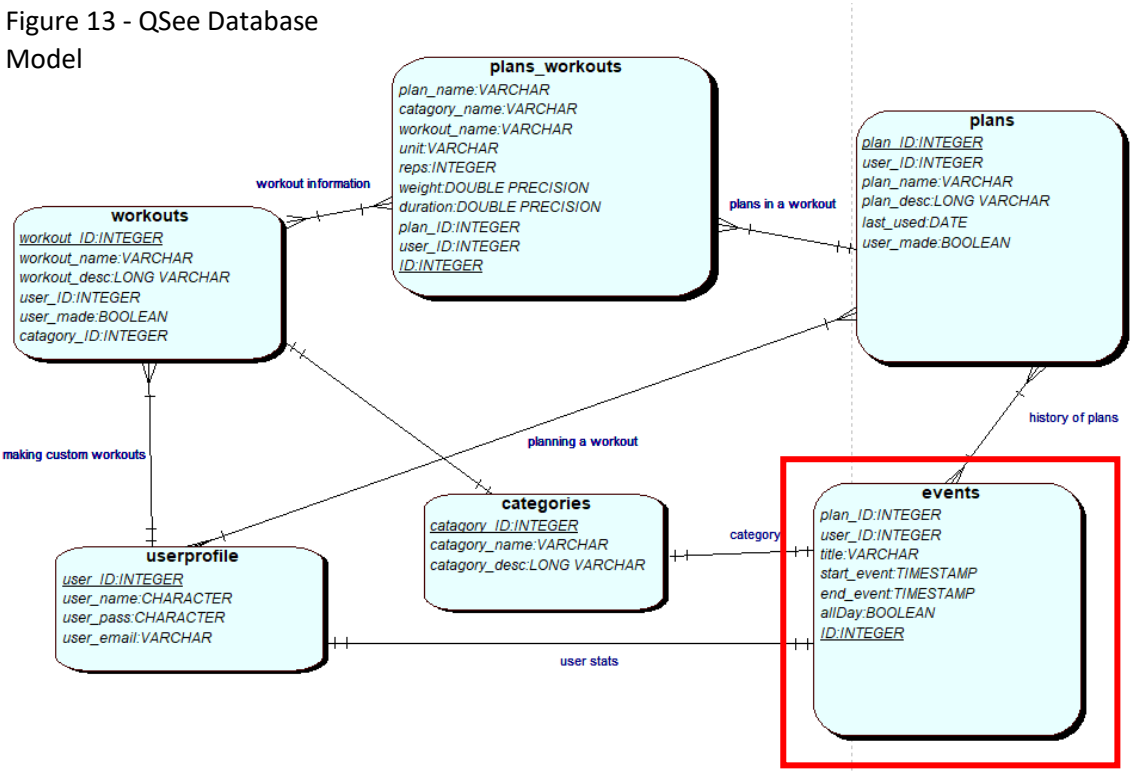
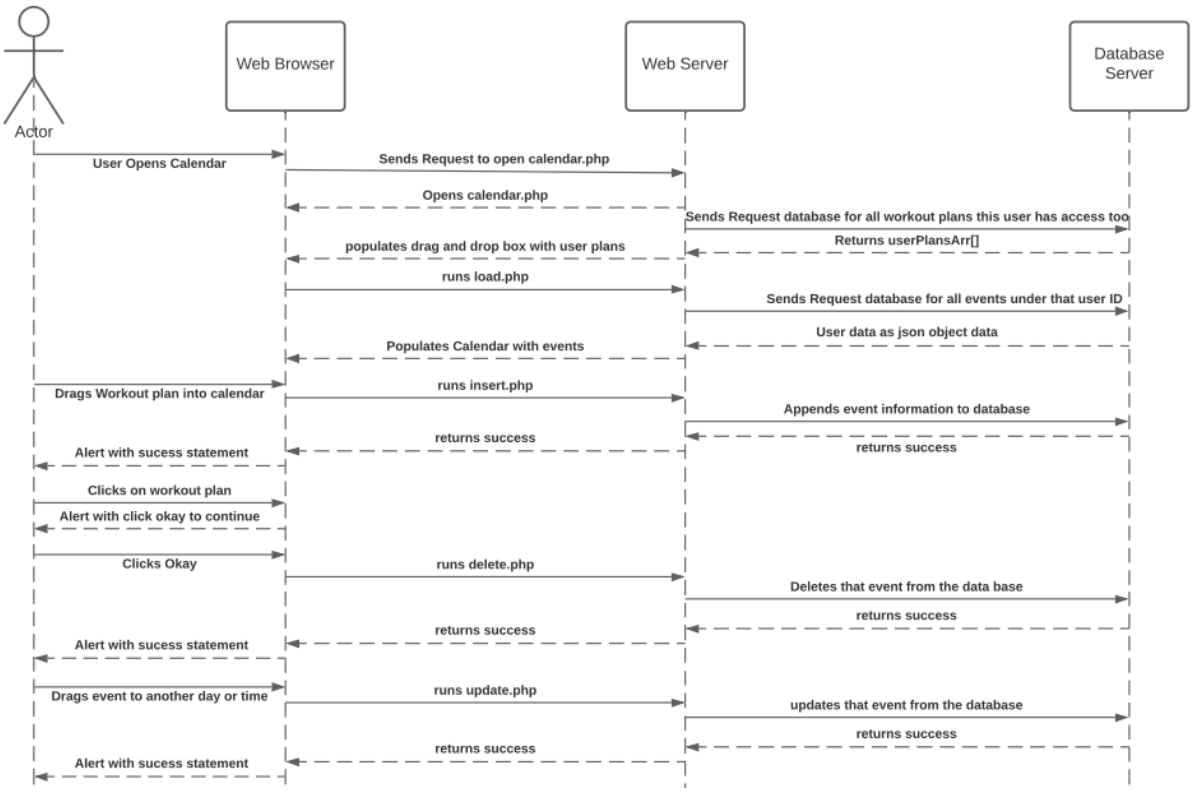


Figure 15 - Schedule a Workout Sequence Diagram



11.3.Implementation

11.3.1.Additional Tools used

- FullCalendar.io - Open source and free to use calendar API (FullCalendar , 2021).

11.3.2.Requirements filled.

Figure 14	
Requirement	Functionality
Schedule Workout Plans (FR 7)	The webpage fulfils this requirement by allowing users to schedule workout plans through the Fullcalendar API.
Delete Workout Plans (FR 8)	By single clicking on a workout plan a user is able to delete workout plans.
Alter Workout Plans (FR 9)	The user is able to alter the times of scheduled workout plans by dragging or resizing them.

11.3.3.Problems and lessons learnt.

1. A confusing issue I came across when attempting to integrate the full calendar with my database was the format of dates, both because they weren't formatted in the same way as my database, and secondly, because it treated an all-day event as if the starting and ending timestamp were the same but, did not output an end date. This meant when trying to save an error was displayed saying no event.end. I fixed this by setting end to the same as the start if no end were given.
2. The calendar adds ghost events when created new events, the way I fixed this was simply refreshing the calendar.

11.3.4.Breakdown

See Appendix D.3 for Schedule a Workout implementation breakdown.

11.4.Review

11.4.1.Verification

See Appendix E.3 for Verification Table.

11.4.2.User Review

See Appendix for F.3 User Reviews Table.

12.Generating statistics

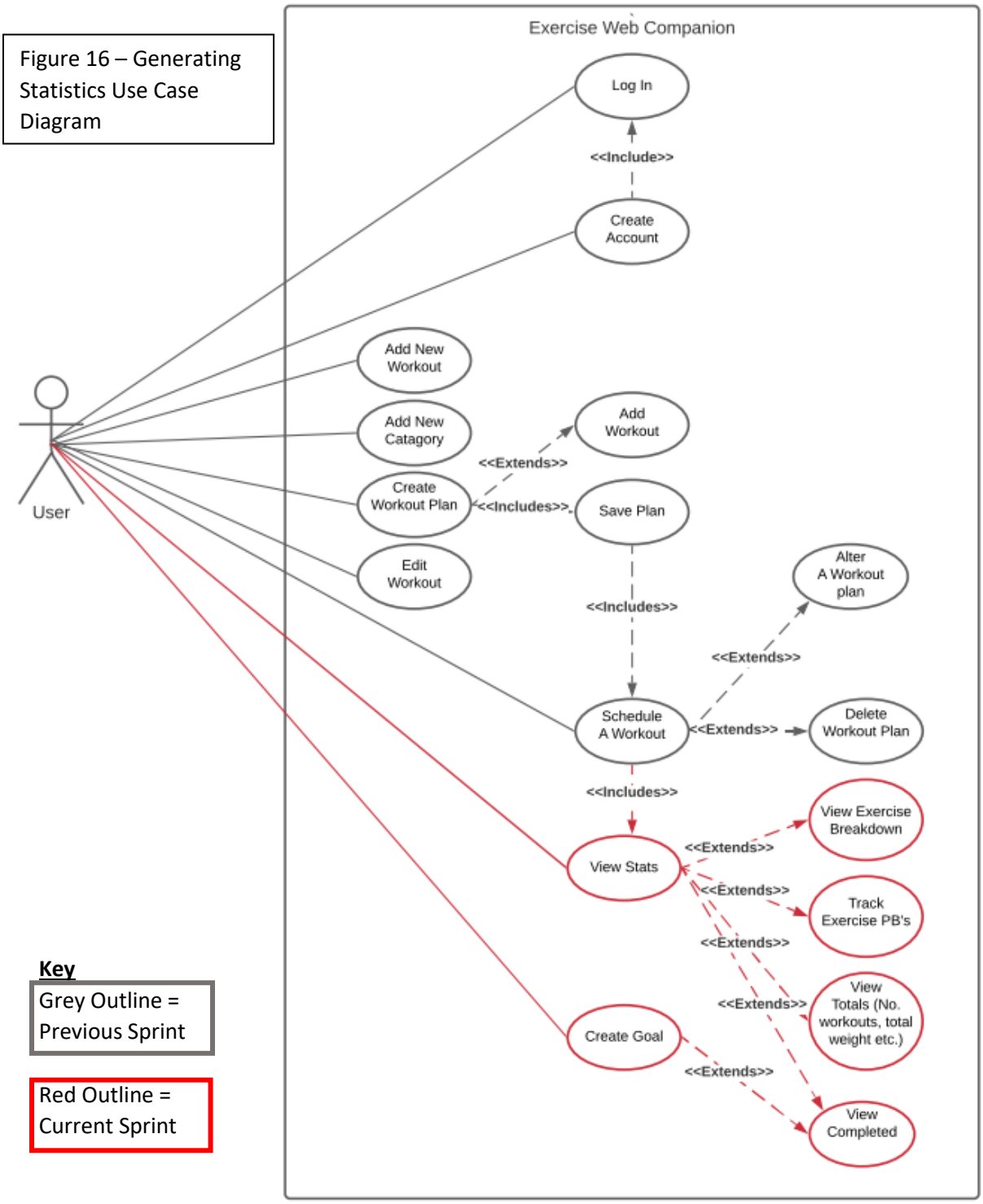
12.1.Requirements

12.1.1.User Stories

See Appendix G.4 for User Stories.

12.1.2.Use Case Diagram

Figure 16 – Generating Statistics Use Case Diagram



12.2.Design

Figure 17 – Updated QSEE Model

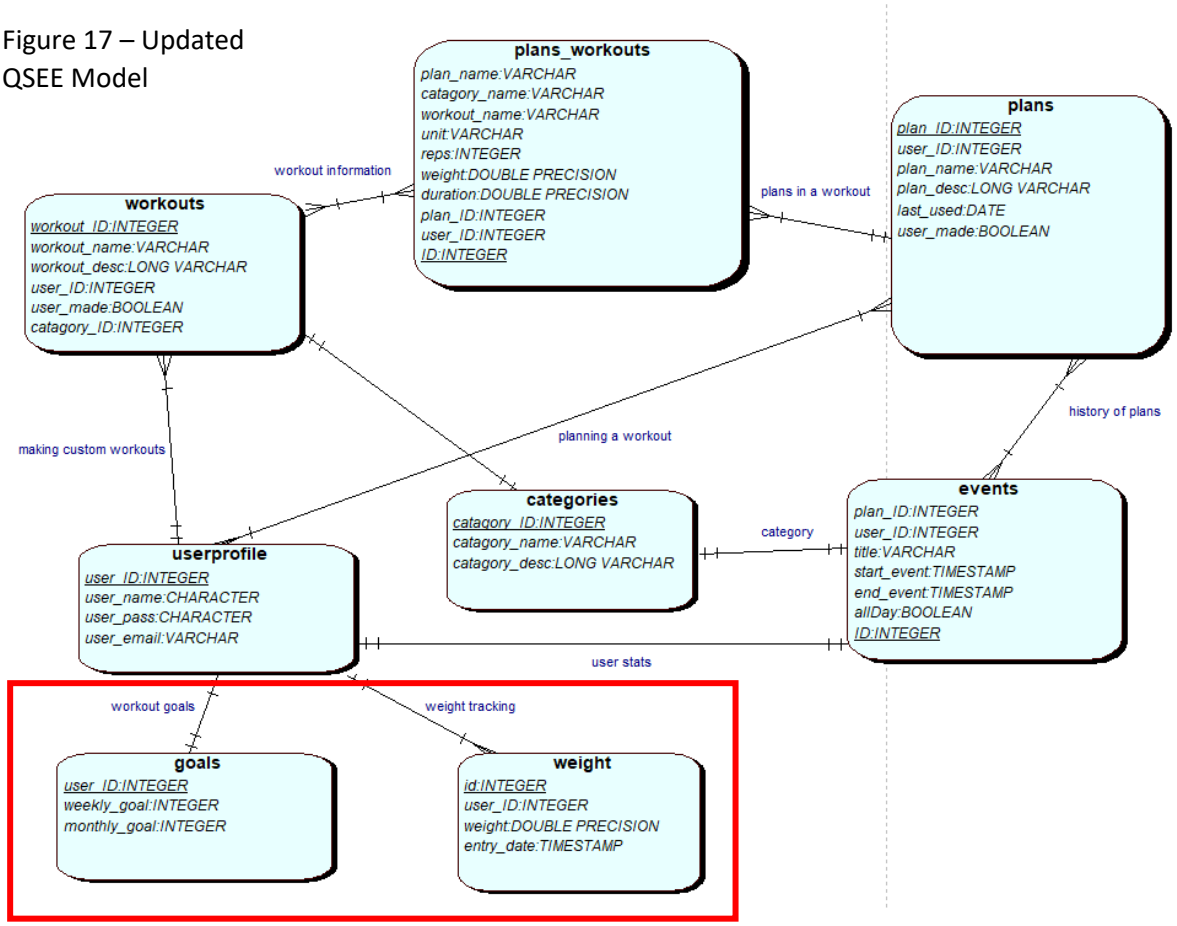


Figure 18 – Sequence diagram Generating Statistics Part 1

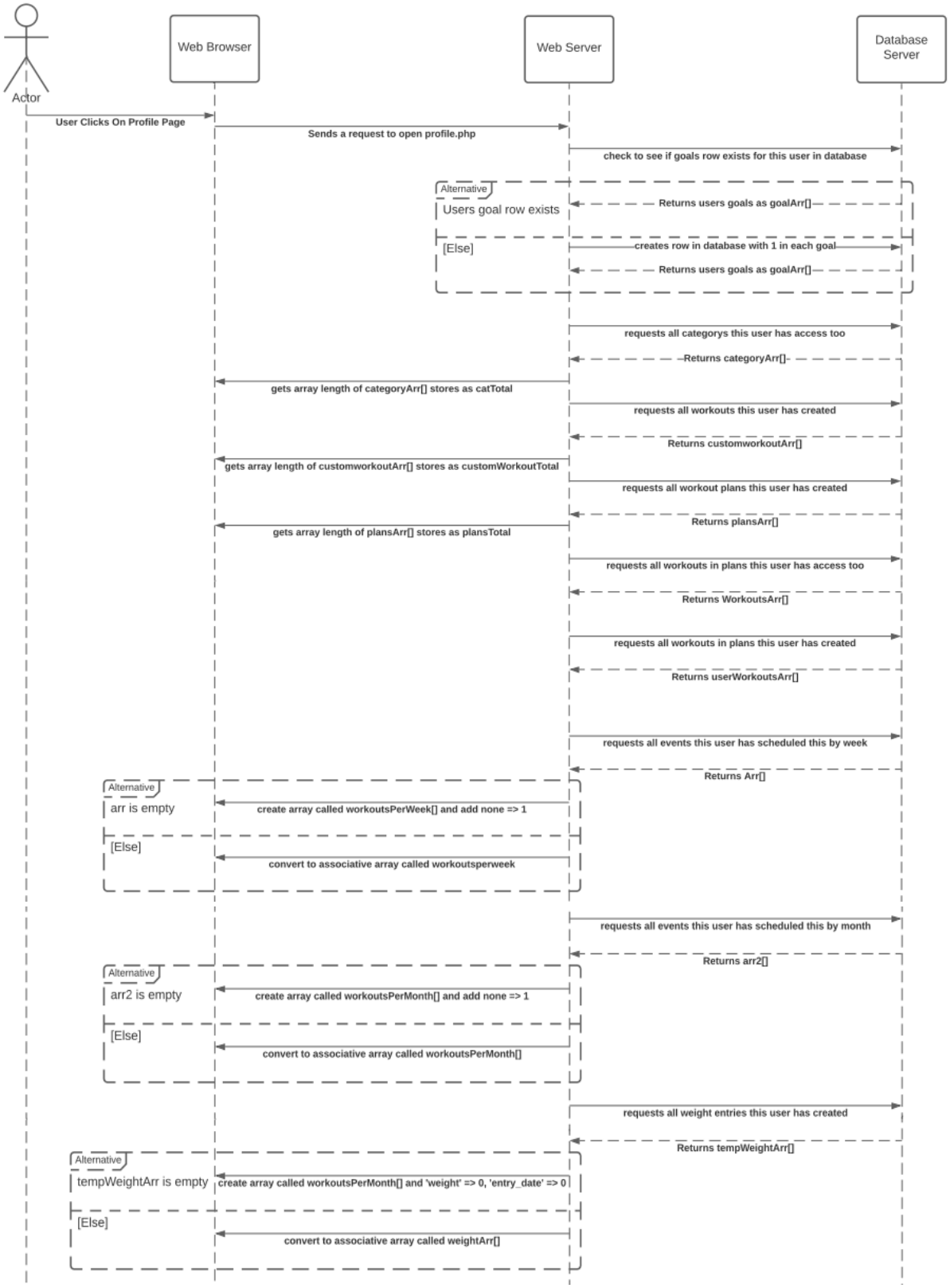


Figure 19 – Sequence diagram Generating Statistics Part 2

12.3.Implementation

12.3.1.Additional Tools used

- **Chart.js** - Chart.Js is an open-source JavaScript API that generates flexible charts (Chart.js, 2021).

12.3.2.Requirements filled.

Figure 20	
Requirement	Functionality
Logging Personal Bests (FR 10)	The profile page fills this requirement by giving a breakdown of the max weight, duration, and Reps for each workout in 3 different time periods.
Seeing Exercise Breakdown (FR 11)	The page shows statistics of category breakdown of workout, favourite and least favourite workout, favourite and least favourite category, totals of all stored variables with useful diagrams.
Weight Tracker (FR 12)	The website provides the user the ability to add their current weight and track it on a graph
Visual Aid's (FR 13)	The website has 3 dynamic charts which display useful data such as category breakdown, workouts completed, and weight.

12.3.3.Problems and lessons learnt.

1. The first problem I faced during this process was, repeatedly having to write out the same function for a given task, I learnt to change my approach and tried to create functions which were more dynamic. This ended up saving a large amount of time.
2. At first, I attempted to just pass a complete array of events to JavaScript and calculate the number of plans completed and therefore workouts manually. This was quite difficult as a did not know how to define calendar weeks. Instead, I opted to use SQL and return arrays of events within a given time frame.

12.3.4.Breakdown

See Appendix D.4 for Generating Statistics implementation breakdown.

12.4.Review

12.4.1.Verification

See Appendix E.4 for Verification Table.

12.4.2.User Review

See Appendix F.4 for User Reviews Table.

13.Content Aggregation

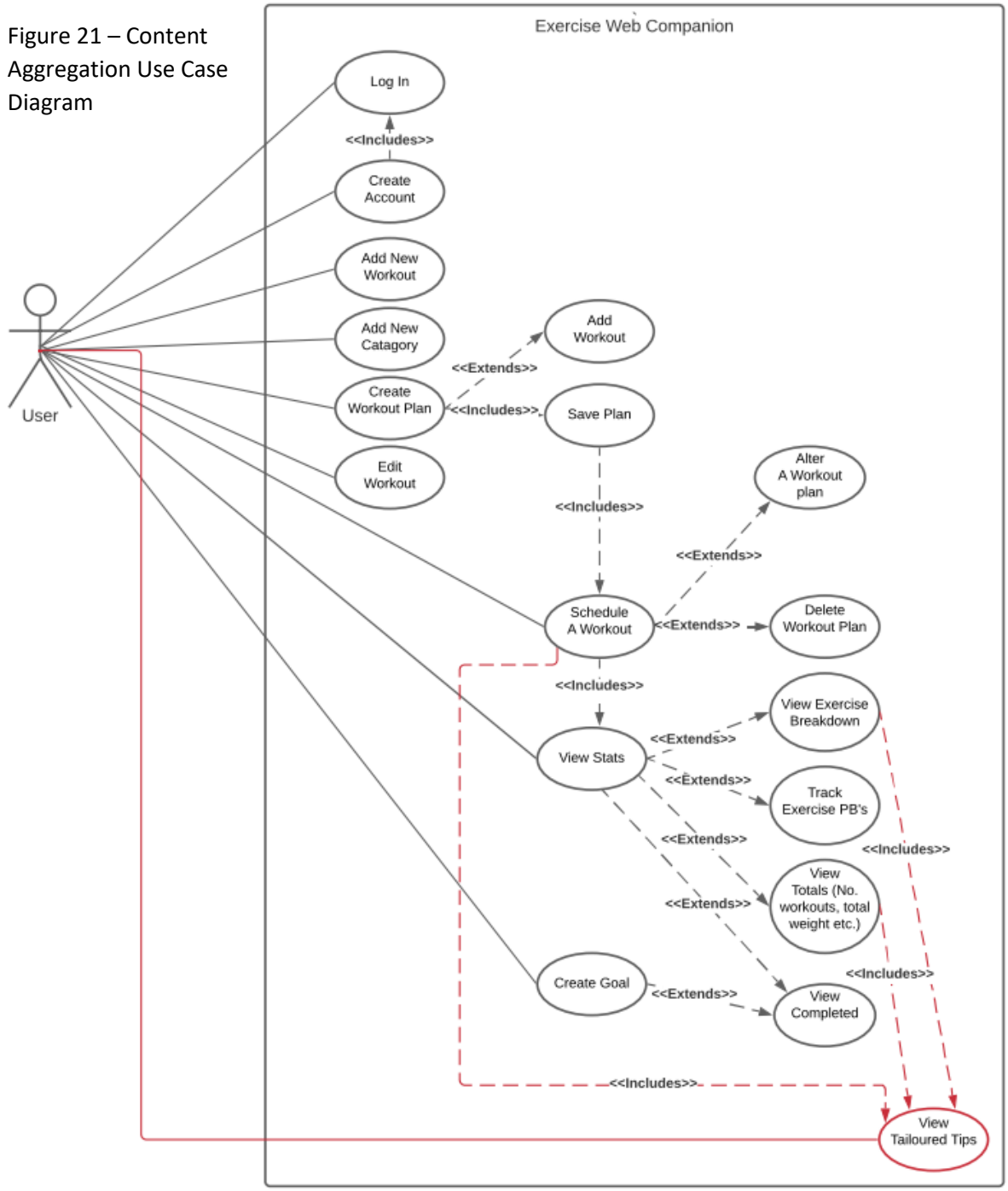
13.1.Requirements

13.1.1.User Stories

See Appendix Error! Reference source not found. **for User Stories.**

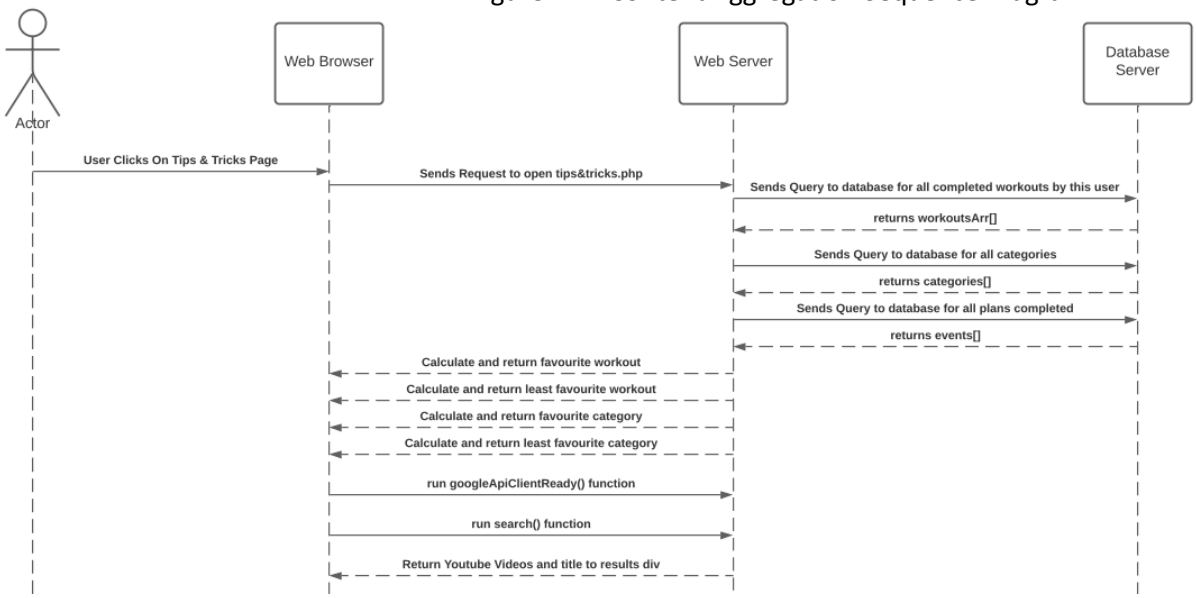
13.1.2.Use Case Diagram

Figure 21 – Content Aggregation Use Case Diagram



13.2.Design

Figure 22 – Content Aggregation Sequence Diagram



13.3.Implementation

13.3.1.Additional Tools used

- **YouTube Data API -** The YouTube API allows developers to access video statistics and YouTube channels data (Google, 2021)

13.3.2.Requirements filled.

Figure 23	
Requirement	Functionality
Tailed Personalised Tips (FR 14)	When the user loads the tips tricks page, it displays tailored YouTube Videos based on their favourite and least favourite, categories and workouts.

13.3.3.Problems and lessons learned.

1. I found it difficult to understand how the API worked since you have to initialise it before the functions for pulling data are loaded. My work around for this was to call the "googleApiClientReady" function separately.

13.3.4.Breakdown

See Appendix D.5 for Generating Statistics implementation breakdown.

13.4.Review

13.4.1.Verification

See Appendix for E.5 Verification Table.

13.4.2.User Review

See Appendix for F.5 User Reviews Table.

14.Quality Assurance

In this section I will be reviewing the web application against the non-functional quality assurance requirements defined in section 5.3.

Figure 24		
NFR & Name	Test Case	Results
NFR1 Security	All passwords must be kept private and hashed to prevent a data breach.	All Passwords are Stored as a one-way hash, and none are exposed in any function or process. See below how passwords are stored in the database. testcase $2y$10$qor193q8IAQxrekkOLGDP.N0Mm4EVKsnUFzJG74/p6Y... testcase@testcase.com
NFR2 Reliability	The website must have the same loading speed and performanc e over the course of 1 week.	From 05/04/21 to 12/04/21 all pages had the same loading time aside from tips and tricks, this varied due to the speed of the YouTube API which unfortunately is not within my control.
NFR3 Usability	The user must be able to intuitively create a new "chest plan" with 5 workouts, save it, and then schedule it for today within 5 minutes.	During the validation process a random user was selected to perform this test before completing the validation process. This meant it was a blind test. They were successful in completing the test with 5 minutes, their time was 4:32 seconds.
NFR4 Efficiency	Each page should load within 5 seconds.	All page's load within 5 seconds, the create account, login, home and calendar pages all load within 1 second. The profile page due to the number of calculations and rendering the charts from the charts.js API took slightly over 1 second and finally the tips and tricks page loaded within 4 seconds due to the YouTube API.
NFR5 Rebuild	The application must be able to be rebuilt on a new server	It took 7 minutes 35 seconds to rebuild the website, it should be noted that the website was recreated on XAMPP and using PHPMyAdmin which greatly improved this speed of this process.

	within 10 minutes of acquiring the files.	
NFR6 Compatibility	The user must be able to intuitively create a new "chest plan" with 5 workouts, save it, and then schedule it for today within 5 minutes on 3 different types of hardware, a desktop, laptop, and smartphone.	The website performs well on a laptop and desktop computer, however on the mobile it can be difficult to navigate through some of the interfaces due to the CSS not changing for the device type. However, all three devices achieved the goal.

15.Conclusion

I will begin by describing my general thoughts of the choice of methodology and the advantages and disadvantages. Firstly, the choice to use an agile framework and specifically scrum was the right decision for this project. It allowed the project to be extremely flexible due to the tight time restraints, organising the functional requirements using MoSCoW came with huge benefits towards the end of the project, as it became clear that I did not have enough time to complete the last two modules. Identifying the core requirements ensured that the must crucial functionality was completed in time and the nature of scrum and it's sprints gave the project a logical flow and consistency.

However, there were some disadvantages due to the nature of completing the project through sprints, at the beginning and middle portions of the project it was difficult to fully understand the scale of the task at hand and sometimes it was difficult to understand where dependencies between requirements and modules lay. If I were to repeat the project again, I would better define and plan out the core functionality of the project this would help me better understand the timescale of creating certain pieces of functionality rather than constantly having to revise my plan.

In terms of the project itself I would argue that it was a success. I was sceptical at first that I would be able to complete all the functions necessary to make it a fully-fledged product. When initial planning started, I had worries that perhaps my technical skills in some of the technologies used were not up to the level required to complete the project however, with careful choice of what software and tools I used and ensuring that they were all well documented and used, I was able to teach myself during the process. From these activities I learnt a great deal regarding, JavaScript, front-end website design and core fundamentals surrounding better practices when programming. Through the review sections of each sprint the project did achieve its requirements, which in tern makes it a success.

If there wasn't such a tight deadline, I would have liked to improve some of the visual aspects, this was noted in almost all of the user review sections as a point of improvement. From watching other users attempt to operate the website unguided, I feel some of the choices in the interface design aren't intuitive and require some prompting to full grasp. Additionally, although they weren't a point of focus, the visual aspects of the website could be greatly improved. Once again due to the time restraint some of the styling was lacked continuity and sometimes made it difficult for the user to clearly read text. Consolidating all styling into 1 CSS file would greatly improve the continuity of the design and spending more time researching user interfaces would better equip myself for the design process.

16.References

Apache Friends, 2021. *XAMPP.* Unknown: GNU General Public License.

Baker, C., 2021. *Obesity Statistics,* London: House of Commons Library.

British Standards Institution, 2001. *ISO/IEC 9126 Software engineering.* London : British Standards Institution.

Chart.js, 2021. *Chart.js.* N/A: Chart.js.

Cooke, K., 2018. *37% of people in the UK never exercise or play sport.* [Online]
Available at: https://www.kantar.com/uki/inspiration/sport-leisure/37-per-cent-of-people-in-the-uk-never-exercise-or-play-sport/
[Accessed 12 January 2021].

CURRY, D., 2020. *Strava Revenue and Usage Statistics (2020).* [Online]
Available at: https://www.businessofapps.com/data/strava-statistics/#:~:text=Strava%20currently%20has%2055%20million,adds%20one%20million%20every%20month.
[Accessed 12 January 2021].

Daily Strength, 2020. *Gym Workout Tracker & Planner for Weight Lifting.* [Online]
Available at:
https://play.google.com/store/apps/details?id=com.anthonyng.workoutapp&hl=en_GB&gl=US
[Accessed 1 November 2020].

Fitness Buddy , 2020. *Fitness Buddy : 1700 Exercises.* [Online]
Available at:
https://play.google.com/store/apps/details?id=com.skyhealth.fitnessbuddyandroid&hl=en_GB&gl=US
[Accessed 1 November 2020].

FullCalendar , 2021. *FullCalendar.* Unknown: FullCalendar LLC.

Google, 2021. *YouTube Data API.* San Bruno:
https://developers.google.com/youtube/documentation.

Jewell, T., 2020. *The Best Fitness and Exercise Apps of 2020.* [Online]
Available at: https://www.healthline.com/health/fitness-exercise/top-iphone-android-apps#map-my-run
[Accessed 1 November 2020].

Lucidchart, 2021. *Lucidchart.* [Online]
Available at: https://www.lucidchart.com
[Accessed 5 April 2021].

NHS, 2019. *Statistics on Obesity, Physical Activity and Diet, England, 2020,* London: NHS.

Nike , 2002. *Nike Training Club—Home workouts and fitness plans.* [Online]
Available at: https://play.google.com/store/apps/details?id=com.nike.ntc&hl=en_GB&gl=US
[Accessed 1 November 2020].

phpMyAdmin, 2021. *phpMyAdmin.* Unknown: GNU General Public License 2.

Rehkopf, M., 2021. *2021.* [Online]
Available at: https://www.atlassian.com/agile/project-management/user-stories
[Accessed 18 2 February].

Statista Research Department, 2018. *Most popular health and fitness apps in the United States as of May 2018, by monthly active users.* [Online]
Available at: https://www.statista.com/statistics/650748/health-fitness-app-usage-usa/#:~:text=With%20the%20growing%20usage%20of,in%20tracking%20their%20health%20digitally.&text=During%20the%20first%20quarter%20of,store%20amounted%20to%2037%2C143%20apps.
[Accessed 20 April 2021].

Sublime Text, 2021. *Sublime Text 3.* Unknown: Sublime Text.

Target Internet, 2021. *AN INTRODUCTION TO AGILE WORKING.* [Online]
Available at: https://www.targetinternet.com/an-introduction-to-agile-working/
[Accessed 18 February 2021].

Under Armour, 2020. *Map My Run by Under Armour.* [Online]
Available at:
https://play.google.com/store/apps/details?id=com.mapmyrun.android2&hl=en_GB&gl=US
[Accessed 1 November 2020].

Appendix

A. Appendix A

A.1. Research of Other Fitness Apps

Map My Run (Jewell, 2020)

- **iPhone rating:** 4.8 stars
- **Android rating:** 4.6 stars
- **Price:** Free

"Map my run" is an app owned by under amour, it primarily focuses on running however offers some, limited, workout plans for other exercise groups such as CrossFit and yoga (Under Armour, 2020).

Interesting Features:

- A mapping Tool for runs which can save routes, also suggests routes.
- Custom Profile settings.
- The ability to add friends.
- Linkable with Facebook.
- Goals.
- In app videos.

Poor Features:

- Overall, the app isn't very responsive, it uses a variety of menu types which could be confusing to new users.
- It advertises a nutrition section however that is a separate app which you are required to download.
- Some fonts are difficult or annoying to read due to their block capital and bold design.
- The features are well thought out it's the design which lets this product down.

Fitness Buddy (Jewell, 2020)

- **iPhone rating:** 4.8 stars
- **Android rating:** 4.1 stars
- **Price:** Free

"Fitness Buddy" markets itself as a virtual personal trainer and nutritionist, it possesses a wide variety of different exercise and plans (Fitness Buddy , 2020).

Interesting Features:

- Fitness buddy has a very intuitive menu system which is helped by a short tutorial upon opening the app for the first time.
- It suggests possible workouts.
- You are able to add custom work outs and plans (If you pay)
- It suggests nutrition advised which is informed by an in-app test.

- Has visual aids on how particular exercises are performed.

Poor Features.

- The app aggressively markets its premium features to the point where mid workout it will pop up with a 30 second screen asking you to purchase.
- Most crucial features such as creating custom work out plans are behind a pay wall.
- Does not contain goals.

Nike Training Club (Jewell, 2020)

- **iPhone rating:** 4.9 stars
- **Android rating:** 4.1 stars
- **Price:** Free

Nike Training Club is an app made by Nike, it is a family-friendly workout app with nearly 200 different workouts that let you do strength, cardio, yoga, and much more without needing to go to the gym or use any equipment (Nike , 2002).

Interesting Features:

- The App is very well organised and is very intuitive.
- Premade workouts have accompanied gifs which make it easy to see what the plan contains.
- Milestones, and visual rewards are a good incentive for motivation.
- Good progress tracking

Poor features:

- Not obvious how to log your own custom work out.
- Clickbait articles are a strange feature for a fitness app.
- Workouts are not group very well.
- Not easy for beginners to understand what is going on due to poor organisation and labelling.
- Advice is not tailed to the user.

Daily Strength (Jewell, 2020)

- **iPhone rating:** 4.7 stars
- **Android rating:** 4.6 stars
- **Price:** Free

Daily Strength is a gym workout planner and workout tracker that makes it easy to log your weightlifting workouts. Create your own gym workout plans and routines by selecting from a list of exercises such as bench press, barbell squat, deadlift, and a variety of dumbbell, barbell, and Olympic exercises. The workout tracker also contains built in gym workout plans that you can easily get started on (Daily Strength, 2020).

Interesting Features:

- Sets goals immediately.
- Good visual indicators of days completed and how close you are to your goal.

- Personal coach feature is well made and good for new users who are new to working out.
- Good visual indication of progress and tracking what areas need improvements.
- Good clear statistics on workouts based on week/month/year.
- Good visual indication of what workouts effect what muscles groups.
- Easy to make a custom workout once you are within that page.

Poor features:

- Mainly focuses on strength training.
- Difficult to see where to add plans at first glance.
- Terminology could be to be complicated for new users.
- Not enough options for running/cardio.

Suggested Features:

- Progress photos
- Adding notifications for working out
- YouTube videos for guidance
- Audio for guidance
- A chatbot for asking questions to.

B. Appendix B

B.1. Primary Research Plan

1. How often do you exercise?
2. What sort of exercise do you part take in? (Select as many as appropriate to you)
3. What is your biggest obstacle to working out?
4. If you do not exercise or when you started exercising, did you find it confusing and difficult to understand how to exercise?
5. Would a website that gives you useful tips and provides reminders encourage you to exercise?
6. Would visual indicators of progress, such as charts, provide motivation for regularly exercise?
7. If you do exercise, do you use any app or website to track progress?
8. Have you ever used any of these exercise apps?
    - Under Amour MapMyRun
    - Android Fitness Buddy
    - Nike Training Club
    - Daily Strength
    - Strava
    - None of the above
9. Are there any key features you liked?
10. Where do you search for information regarding exercise?
    - YouTube
    - Google
    - Books
    - Friends + Family
    - Other

11. How would you rate these features for an exercise web companion? (10 = Highest 1 = Lowest)
    - Logging Workouts (Creating Custom workout plans)
    - Pre-set workouts to follow.
    - Tracking progress such as personal bests, weight, etc
    - Tailored tips to your workouts
    - Integration with google calendar for reminders and scheduling workouts.
    - Visual representation of where you ran.
    - Suggested running routes.
12. Rank these features (1 = Highest 10 = Lowest)
    - Logging Workouts (Creating Custom workout plans)
    - Pre-set workouts to follow.
    - Tracking progress such as personal bests, weight, etc
    - Tailored tips to your workouts
    - Integration with google calendar for reminders and scheduling workouts.
    - Visual representation of where you ran.
    - Suggested running routes.
13. Do you use google calendar to plan and organise your life?

B.2. Processed Data from Survey

B.2.1. How often do you exercise?

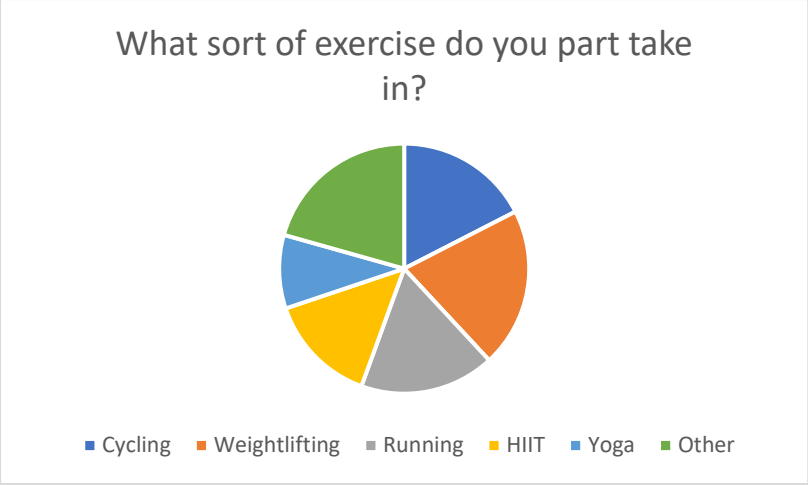
Not at All	5
1-3 Days A week	9
3 - 5 Days a week	10
6+	3

B.2.2. What sort of exercise do you part take in?

Cycling	11
Weightlifting	13
Running	11
HIIT	9
Yoga	6
Other	13

Other specified exercises:

- Walking
- Calisthenics
- Football
- Rowing
- TRX
- Martial Arts

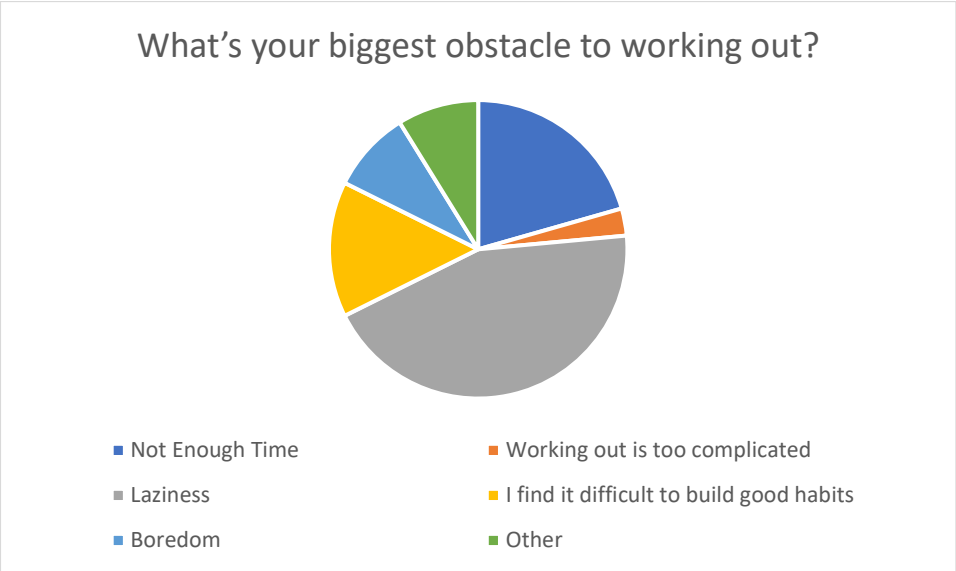


B.2.3. What's your biggest obstacle to working out?

Not Enough Time	7
Working out is too complicated	1
Laziness	15
I find it difficult to build good habits	5
Boredom	3
Other	3

Other Reasons Given:

- Prefer working out with others in person.
- Disability

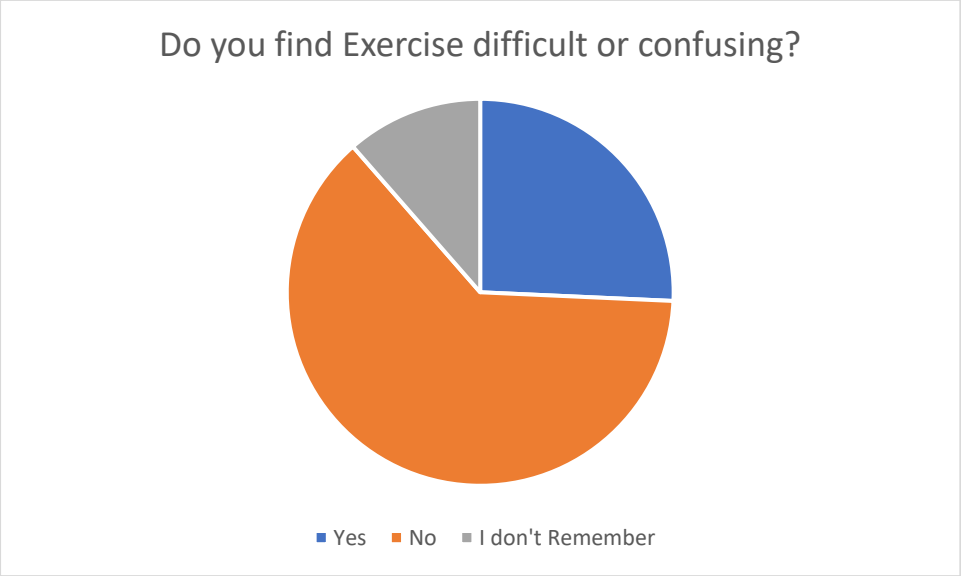


B.2.4. Did you or do you find Exercise difficult or confusing?

Yes	9
No	22
I don't Remember	4

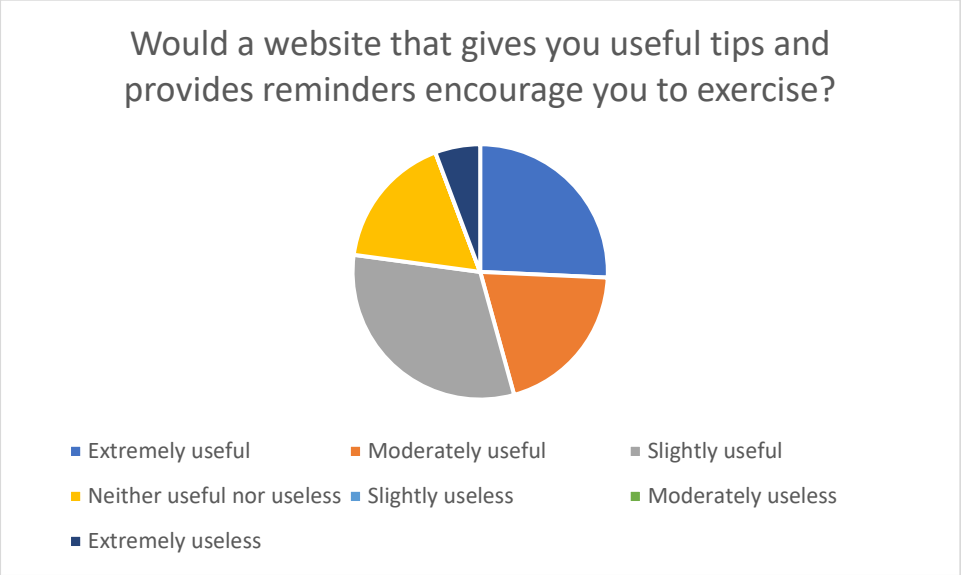
Reason given for confusion:

- I did not know what the best exercise was for me to be focusing on.
- Beginning weightlifting in the gym can be difficult if you don’t have anyone that is able to tell you the best exercises to do for certain muscles.
- It is not always easy to know how to do the exercises efficiently.
- I did not know how to use the gym properly or how to enjoy it.
- Deciding which exercises where best, how long for, how many reps etc
- I found it difficult until I found workout guides on YouTube (like for gym equipment)
- It was difficult to know what the best exercise for my circumstances was.
- Unsure if doing the right exercises.
- Have no balance due to spinal injury.



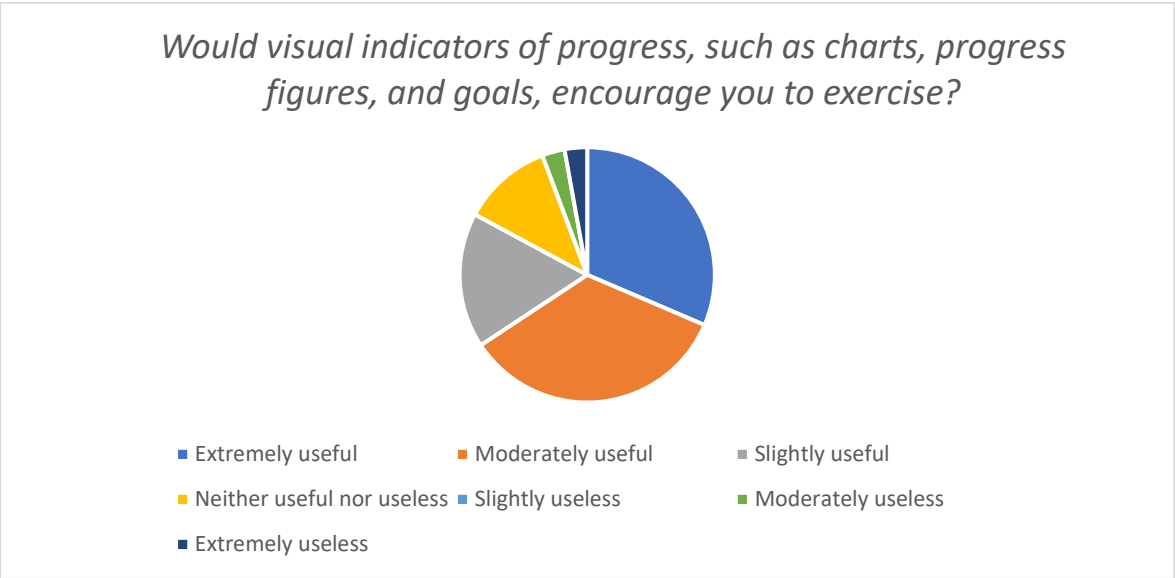
B.2.5. Reminders to encourage you to exercise.

Extremely useful	9
Moderately useful	7
Slightly useful	11
Neither useful nor useless	6
Slightly useless	0
Moderately useless	0
Extremely useless	2



B.2.6. Use visual indicators of progress.

Extremely useful	11
Moderately useful	12
Slightly useful	6
Neither useful nor useless	4
Slightly useless	0
Moderately useless	1
Extremely useless	1



B.2.7. Have you ever used any of these exercise apps?

Strava	2
Daily Strength	2
Under Amour MapMyRun	1
Android Fitness Buddy	2
Nike Training Club	4
Other	3

Other Exercise Apps Specified:

- Places Locker App
- Gymshark
- Aflete
- Couch to 5k

Features that were liked:

- The ability to track goals.
- I liked the wide range of exercises w/ pictures so that I could try new methods myself.
- Leader boards; competing with friends.
- You can plan a workout and do it in time with the app.
- Step by step exercises and workout routines
- Motivation and reminders are really useful.

B.2.8. Where do you find information regarding exercise?

YouTube	18
Facebook	0
Google	16
Friends or Family	9
Books	2
Other	4

Other sources:

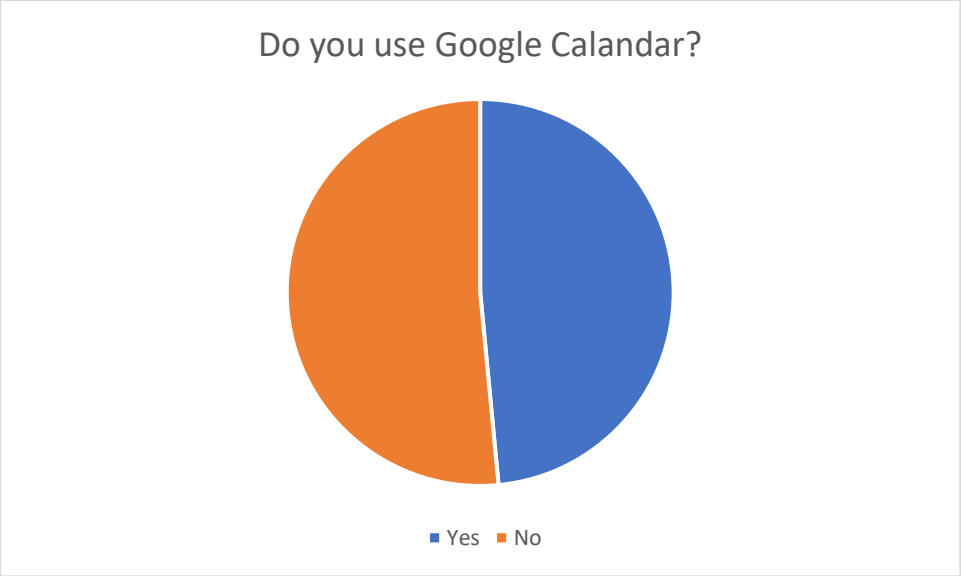
- University
- PT Instructors
- Classes

B.2.9. Rating and ranking of features.

	Average Rating for each feature	Average Ranking
Logging Workouts (Creating Custom workout plans)	7.17	2.69
Pre-set workouts to follow	6.77	2.66
Tracking progress such as personal bests, weight, etc	7.97	2.81
Tailored tips to your workouts	6.7	3.91
Integration with google calendar for reminders and scheduling workouts	5.9	4.44
Visual representation of where you have run	6.77	4.5
Suggested running routes	6.36	

B.2.10. Use google calendar or any other calendar software.

Yes	16
No	17



C. Appendix C - Ethics

C.1. Ethical Checklist

Ethical Review Checklist for Undergraduate and Postgraduate Modules

Staff and PG research students must not use this form, but should instead, if appropriate, submit a full application for ethical approval to the Faculty Research Ethics Committee (FREC).

Please provide project details and complete the checklist below.

Project Details:

Module name	Software Development Project
Module code	UFCFFF-30-3
Module leader	Steve Battle
Project Supervisor	Mehmet Aydin
Proposed project title	Exercise Web Companion

Applicant Details:

Name of Student	Paul Holt
Student Number	18012589
Student's email address	paul2.holt@live.uwe.ac.uk

	CHECKLIST QUESTIONS	Yes/No	Explanation
A.	Does the proposed project involve **human tissue, human participants, animals, environmental damage, or the NHS.**	No	*If the answer to this is 'No' then no further checks in the list need to be considered.*
B.	Will participants be clearly asked to give consent to take part in the research and informed about how data collected in the research will be used?		
C.	If they choose, can a participant withdraw at any time (prior to a point of "no return" in the use of their data)? Are they told this?		

CHECKLIST QUESTIONS		Yes/No	Explanation
D.	Are measures in place to provide confidentiality for participants and ensure secure management and disposal of data collected from them?		
E.	Does the study involve people who are particularly vulnerable or unable to give informed consent (eg, children or people with learning difficulties)?		
F.	Could your research cause stress, physical or psychological harm to humans or animals, or environmental damage?		
G.	Could any aspects of the research lead to unethical behaviour by participants or researchers (eg, invasion of privacy, deceit, coercion, fraud, abuse)?		
H.	Does the research involve the NHS or collection or storage of human tissue (includes anything containing human cells, such as saliva and urine)?		

Your explanations should indicate briefly for Qs 2-4 how these requirements will be met, and for Qs 5-8 what the pertinent concerns are.

- **Minimal Risk:** If **Q 1 is answered 'No'**, then no ethics approval is needed.
- **Low Risk:** If **Qs 2-4 are answered 'Yes' and Qs 5-8 are answered 'No'**, then no approval is needed from the *Faculty Research Ethics Committee* (FREC). However, your supervisor must approve (a) your information and consent forms (Qs 2 & 3) and (b) your measures for participant confidentiality and secure data management (Q4).
- **High Risk:** If **any of Qs 5-8 are answered 'Yes'**, then you must submit an application for full ethics approval *before* the project can start. This can take up to 6 weeks. Consult your supervisor about how to apply for full ethics approval.

Risk Assessment: Separate guidance on risk assessment can be found on UWE's Health and Safety forms webpage at https://go.uwe.ac.uk/RiskAssessment. If needed, you must complete a Risk Assessment form. This must also be attached to your application for full ethics approval if your project is **High Risk**.

Your supervisor must check your responses above before you submit this form.
Submit this completed form via the *Assignments* area in Blackboard (or elsewhere if so directed by the module leader or your supervisor).

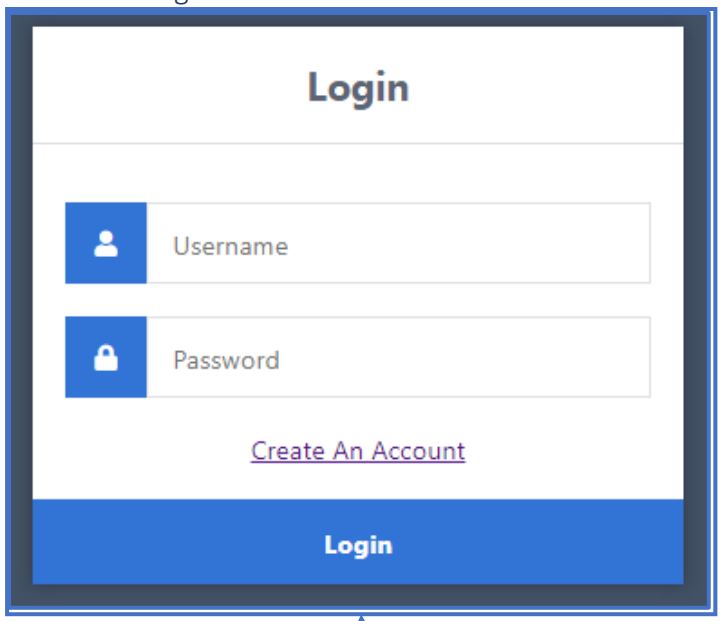
After you have uploaded this form, your supervisor will confirm it has been correctly completed by "marking" it as *Passed*/100% via the *My Grades* link on the Blackboard.

Further research ethics guidance is available at
http://www1.uwe.ac.uk/research/researchethics

D. Appendix D - Implementation Breakdown.

D.1. Creating an Account Breakdown

Creating an Account Breakdown Part 1



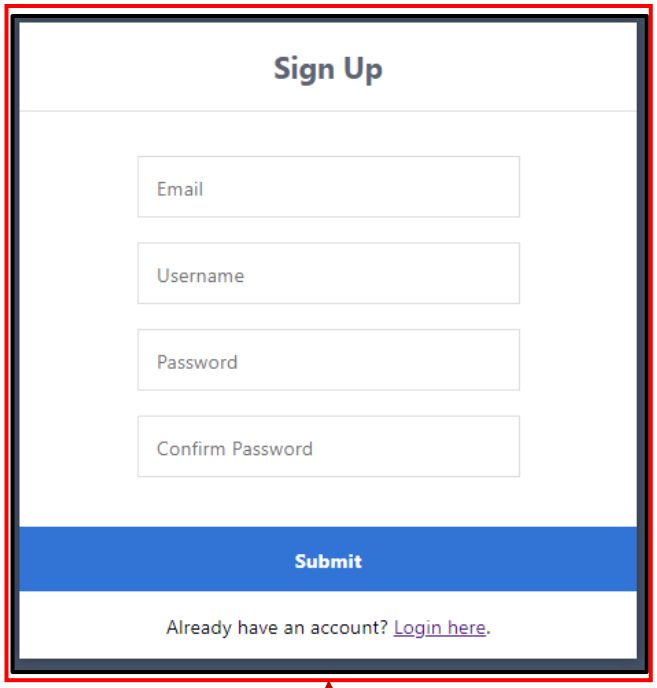
1# The first step of implementation was to create a database based on the design shown in *figure 6.* This database is the backbone of the whole project and is how individual's information will be stored, ranging from password and profile information to individual workouts and plans. Some aspects of the database are subject to change at this point as the project evolves.

The table which will be the primary focus of this section is "userprofile", below shows an example "demo" of how information is stored within the database, displayed using PhpMyAdmin.

user_ID	user_name	user_pass	user_email	fav_workout	fav_workout_plan	calander_enabled
9	demo	$2y$10$by51nU5q9hbDVIKrq.j3PuvLxML0BET1jyleGAZEnfH...	demo3@demo3.com	0	0	0

The visuals were achieved with CSS; however, I won't be going in depth since they're fairly self-explanatory. The body of the document is comprised of a form which contains the input boxes where users provide account information. This information once "login" is pressed is submitted using the Post method to "authenticate.php". This contains the necessary checks to verify a user's login information. In the nature of privacy I used a oneway hash to prevent leaks of private information, I used the PHP function password_verify to check if the password was correct without exposing it. **(FR 1)**

Creating an Account Breakdown Part 2



2# If the "Create An Account" button is pressed the user is directed to the "createprofile.html" which contains an additional form containing input boxes for Email, username, password, and password confirm.

The HTML form once submitted sends the information entered by the user to "createprofile.php" using the POST method, I used this method to ensure password security. Additionally, across any pages passwords aren't made visable to prevent password theft.

Once the server has receieved the POST method a number of validation steps take place including checking if usernames exist, passwords are the correct format and emails are valid and not taken.

Once again to maintain security a hash of the password is used. Along the way the information has been bound to variables within the function "mysqli_stmt_bind_param", If all parts successfully pass validation the variables are inserted into the table userprofile within the database. **(FR 2)**

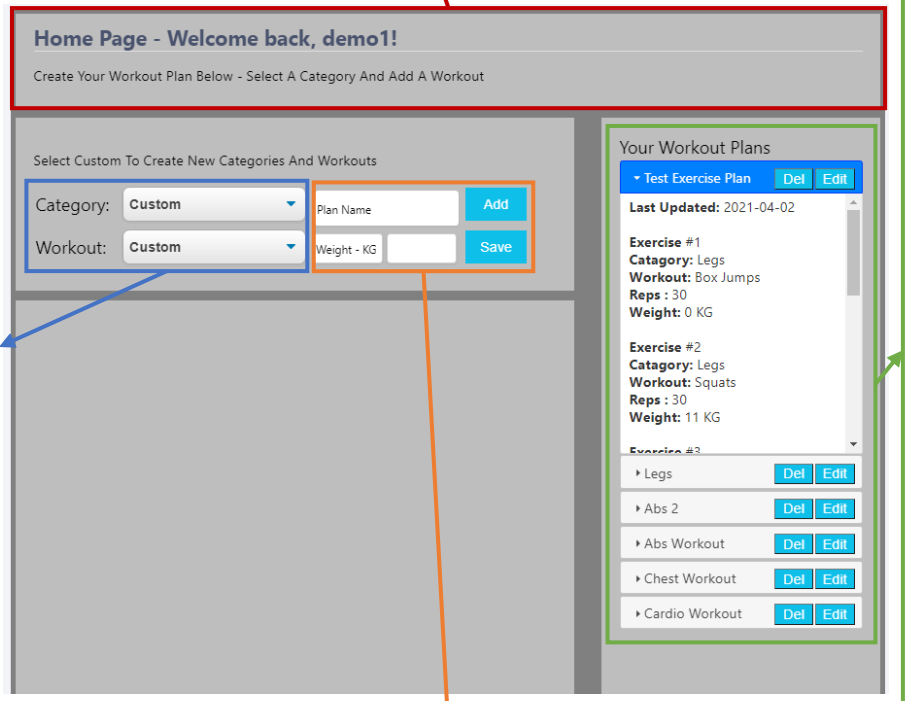
Plan A Workout Breakdown Part 1

#1 All Categories, workouts and plans share the "user_made" and "user_ID" attributes. These attributes coupled with the current user's ID obtained through the session, allow the website to dynamically populate the page with custom categories, plans and workouts which this individual user has created, while also gathering all globally available categories, plans and workouts. This feature is crucial as it makes it possible for users to create custom plans without effecting the experience of others. Below is an example of how the categories are stored, notice how these are globally available categories meaning the user id is null and user_made is 0.

catagory_ID	catagory_name	catagory_desc	user_made	user_ID
1	Custom	Custom category	0	NULL
2	Chest	Working out the chest means working out the pector...	0	NULL
3	Cardio	Cardio exercise simply means that you're doing a r...	0	NULL

#2 Once the data from the category, plans, and workouts tables is gathered from the database it is used to populate the two dropdowns. The two dropdown menus contain global categories and workouts (available to all users) and custom categories (available only to this user). The user is able to select a workout based on the category and name, then enter the weight, reps or duration.

```
//update the workout too
function workoutChange (){
    for(j=0; j<workoutArr.length;j++){

        if (workoutArr[j].workout_name == getWorkoutName()) {
            if (workoutArr[j].repstime == 0) {
                $("#timereps").attr("placeholder","Reps");
            } else {
                $("#timereps").attr("placeholder","Duration");
            };
        };
    };
};
```



#3 This jQuery accordion contains all the workout plans and the workouts within them, it allows the users to click through and edit at will using the edit button. It is implemented in the same way the drop down works only using the json object from the plans query to format and populate the accordion box, performed by the updateAccordion function.

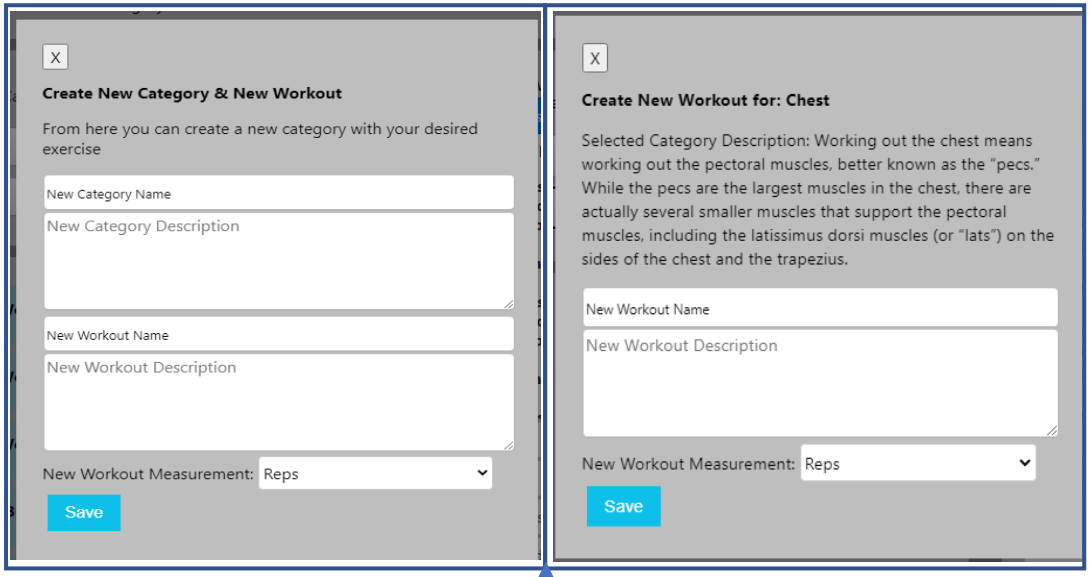
```
for (z=0; z<userWorkoutsArr.length;z++) {

    //if planID is = selected plan, add to string to apply to accordion
    if (userWorkoutsArr[z]["plan_ID"] == plan_ID) {

        //add individual workout list items
        string += "<li> <b>Exercise </b>#";
        string += (count).toString();
        string += "<br><b>Catagory:</b> ";
        string += userWorkoutsArr[z]["catagory_name"];
        string += "<br><b>Workout:</b> ";
        string += userWorkoutsArr[z]["workout_name"];
        string += "<br><b>";
        string += userWorkoutsArr[z]["unit"];
        string += ":</b> ";

        //detect workout unit
        if (userWorkoutsArr[z]["unit"] == "Duration") {
            string += userWorkoutsArr[z]["duration"];
        }else{
            string += userWorkoutsArr[z]["reps"];
        };

        string += " <br><b>Weight:</b> ";
        string += userWorkoutsArr[z]["weight"];
        string += " KG</li><br>";
        count ++;

    };
};
```

#4 The "add" button calls the function "createListElement" which creates a new item in the jQuery list containing the aforementioned information. This list item is dynamically assigned an id alongside the table and buttons which are contained within it. Creating a unique ID which corresponds to all elements within its list item is crucial to allow for the possibility of editing, deleting, and saving while creating dynamic elements *see the Plan A Workout Breakdown Part 3 – list items figure*. The line of code below shows all the values and dynamic ID's being concatenated before being appended to the jQuery sortable.

```
var str = "<table id ='"+temptableID+"'><tr><th><span class='ui-icon ui-icon-arrowthick-2-n-s'></span></th><th>"+catName+" </th><th>"+workoutName+"</th> <th>"+measurement+" :
    </th><th>"+amount+"</th> <th>Weight : </th><th>"+weight+"</th> <th>Kg</th><input class='delbutton' type='button' value='Del' id='"+tempdelID+"'/><input class='copybutton'
    type='button' value='Copy' id='"+tempCopyID+"'/></tr></table><input type='hidden' class='workoutID' value = "+workoutID+" id="+tempHidID+">";
```

The jQuery list allows for users to change the order of list items while in runtime, this allows for a fluid experience. The save button creates a json object containing all the information including the order of workouts. This json object with the workout name is then sent to "submit.php" using the post method, where it is then inserted into the database under the plans table using the users ID from the session. Once a plan has been created it will appear in the jQuery accordion.

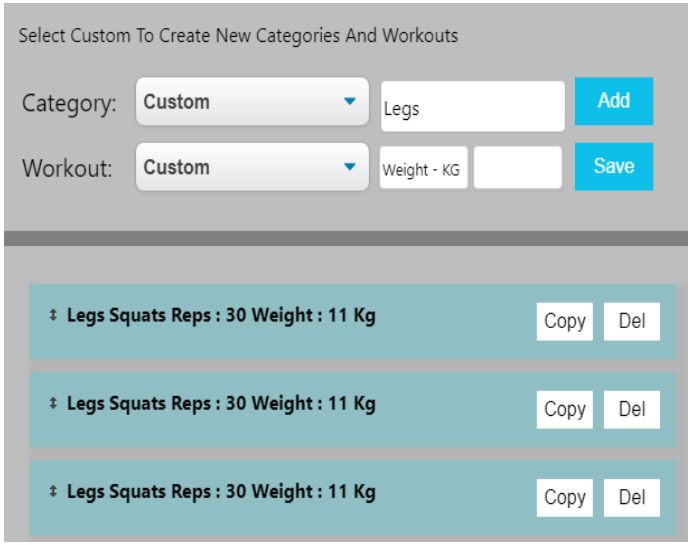
Plan A Workout Breakdown Part 2



If I desired workout or category does not exist within the database, the user has the ability to add one using the custom option in the dropdown menu. If custom is selected in the category dropdown and the "add" button is pressed a popup is opened. Here the user is able to add a new category, workout name, description, and define the unit of measurement (reps or duration). Once save is selected and "createNewCategory.php" is called. The values are posted here and then inserted into the "workouts" and "categories" tables. Alternatively, if only custom is selected in the workouts section and "add" is pressed a similar popup is opened except, now a user can add a new workout for the specific category which is selected using "createNewWorkout.php".

```
//Create Query to sent to db for new cat
$sql = "INSERT INTO categories (catagory_name, catagory_desc, user_made, user_ID) values ('$newCat','$newCatDesc','1', '$userID');";

//Send Query
mysqli_query($con,$sql);

//find last inserted ID
$last_id = $con->insert_id;

//Create Query to sent to db for new workout
$sql = "INSERT INTO workouts (workout_name, workout_desc, user_ID, user_made, catagory_ID, repstime)
values ('$newWorkout','$newWorkoutDesc','$userID', '1','$last_id','$newMeasurement');";

//Send Query
mysqli_query($con,$sql);
```

Plan A Workout Breakdown Part 3 – List Items



Schedule A Workout Breakdown Part 1

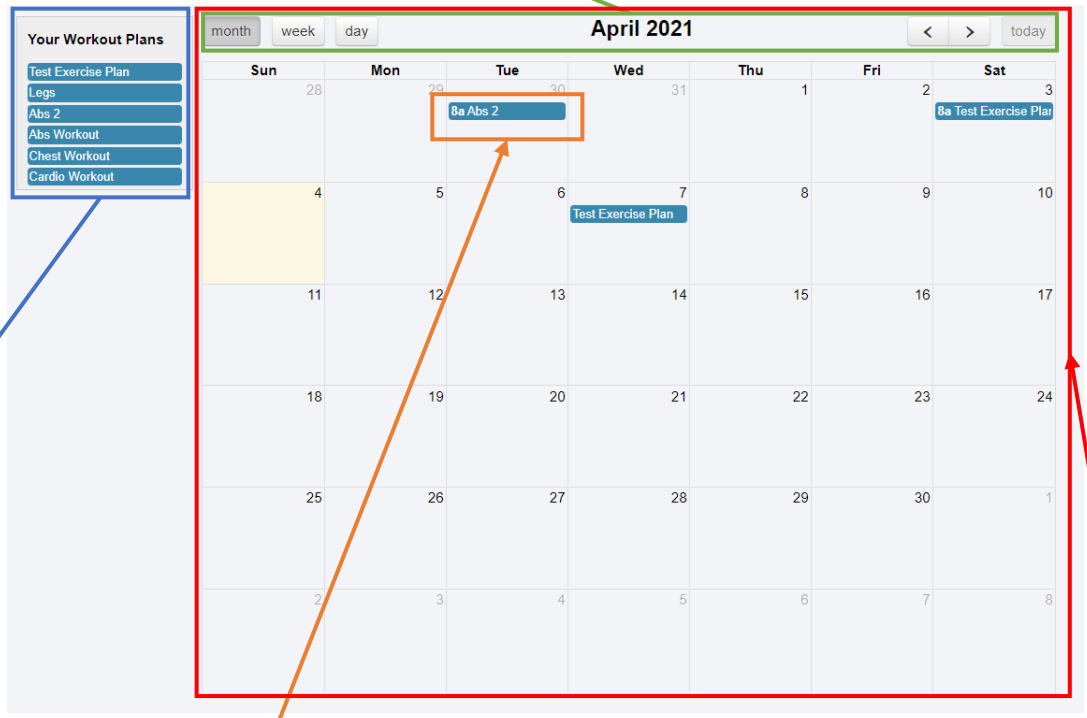
1# The first step was to initialise the calendar through the script tags and calling the "Fullcalendar" method which the desired properties.

```
$('#calendar').fullCalendar({
  header: {
    right: 'prev,next today', //positions the the prev/next button on the right
    center: 'title', //sets the title of the month to center
    left: 'month, agendaWeek, agendaDay' //positions the the prev/next button on the left
  },
```

This module uses the previously unused table within the database, "plans_history", this table serves as storage for all events, however a number of adjustments were made to accommodate Fullcalendar. These includes changing the name to events (this seemed to fit its purpose better), changing the attributes to contain, plan_ID, user_ID, title, start_event, end_event, allDay, ID. See *figure 18.*

2# Since I wanted users to be able to drag and drop workout plans into the calendar, I needed to ensure that the "droppable" property was set to true. Also, I needed to create a widget containing the workout plans available to that user. This was achieved by making a query to the database for all plans had the same user ID and then passing it to JavaScript as a json object. This Json object would then be iterated through and converted into draggable jQuery objects containing data about the user ID, plan ID whether it was an "all day" event, this will become important later. **(FR 7)**

```
title: $.trim($(this).text()),
plan_ID: $(this).attr("id"),
user_ID: user_ID,
allDay: true,
stick: true
```



3# I created the "load" function by first manually entering some data into the database and defining the "events" property of the calendar as "load.php", this means that whenever the calendar is called it would make a request to this php file.

```
events: 'load.php',
```

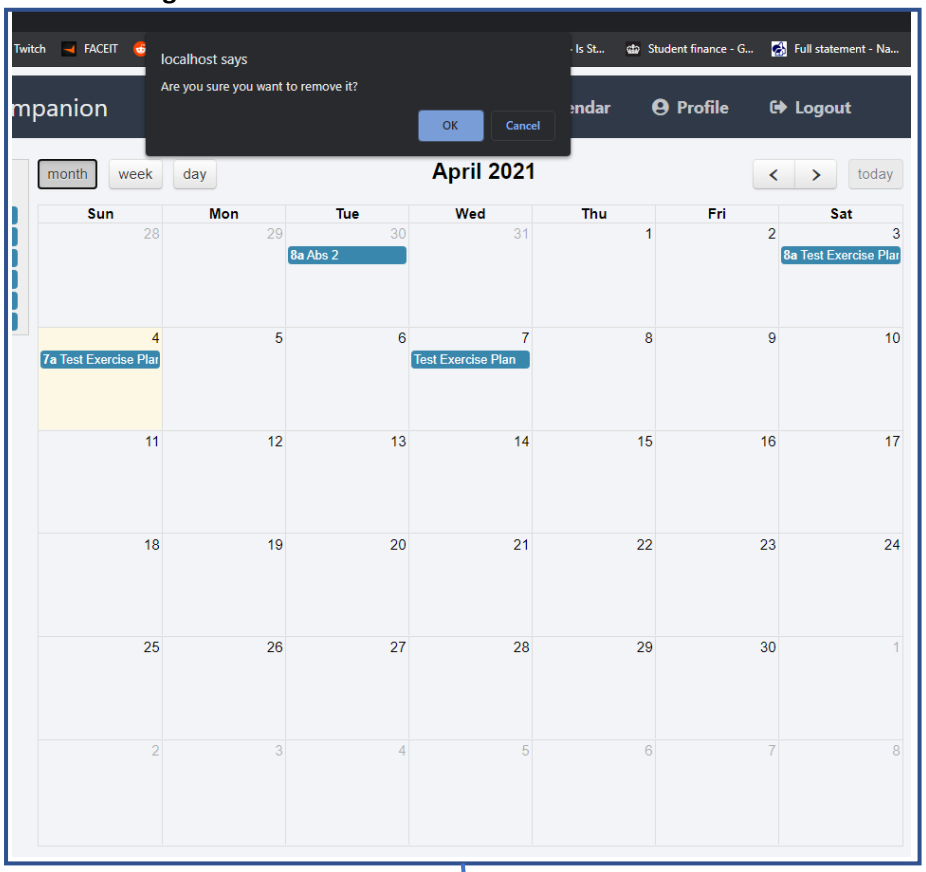
Contained within this file is a SQL query which gets all events in the events table. Once completed a function is called converting the SQL 1 or 0 of the all-day attribute to a Boolean true or false to ensure compatibility. The data then is placed into a json object and echoed; this passes it to Fullcalendar to be displayed. **(FR 7)**

```
foreach($result as $row)
{
 $data[] = array(
  'id'    => $row["id"],
  'user_ID' => $row["user_ID"],
  'title'   => $row["title"],
  'start'   => $row["start_event"],
  'end'   => $row["end_event"],
  'plan_ID'   => $row["plan_ID"],
  'allDay' => boolean($row["allDay"])
 );
}

echo json_encode($data);
```

4# Since the calendar has the "droppable" property it opens up the "eventRecieve" function. This method is called upon receiving a draggable element. First, it is required we take the event timestamp (start and end). Next, we need to convert the date and timestamp into the correct format. This is achieved using the moment function. Finally, we specify the plan id and if the event is all day, this is achieved by reading the data contained within the dropped widget. Then using ajax we post this to the "insert.php" function. Upon success we prefetch the events, reload the calendar and alert a success message. The "insert.php" file reads the array and inserts them into the database. **(FR 7)**

```
$.ajax({
 url:"insert.php",
 type:"POST",
 data:{user_ID: user_ID, title:event.title, start:start, end:end, plan_ID: plan_ID, allDay: allDay},
```

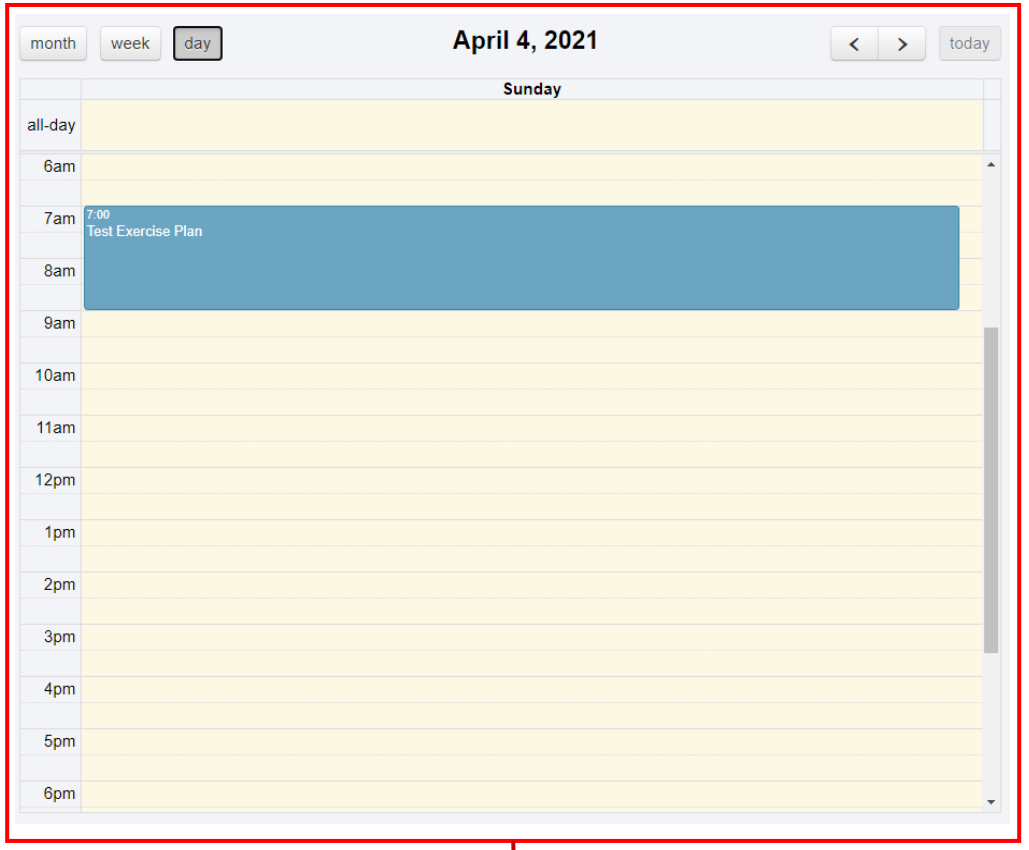
Figure 22 – Schedule A Workout Breakdown Part 2



This is called when a user single clicks on an existing event the "eventClick" function is called. The "eventClick" function contains an alert box which asks if the user wants to delete, if the outcome is true the it posts the id of the event to "delete.php" where a delete query is run using the ID of the event. **(FR 7 + 8)**

```
eventClick:function(event){
 if(confirm("Are you sure you want to remove it?")){
  var id = event.id;
  $.ajax({
   url:"delete.php",
   type:"POST",
   data:{id:id},
   success:function(){
    $('#calendar').fullCalendar('refetchEvents');
    alert("Event Removed");
```

Schedule A Workout Breakdown Part 3



Full Calendar allows the user the ablity to change the calendar view between, month, week and day. In the week and day sections, users are able to change the times of events within 30 minutes. This functionality calls the "eventResize" function. This similar to the insert function gets the end, start, plan_ID, title, id, and whether or not the event is all day then sends it to the "update.php" using the post method through ajax. Once here the script then processes the json object and updates the correct event in the database using the ID. **(FR 7 + 9)**

D.4. Generating Statistics Breakdown

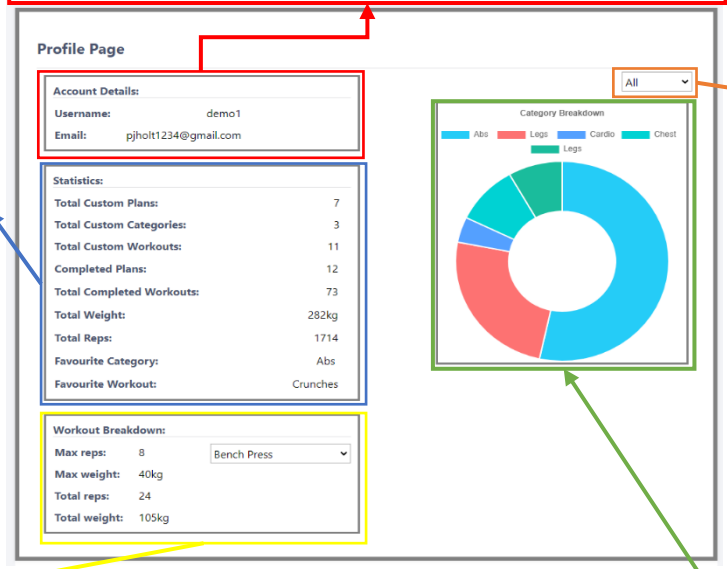
Generating Statistics Breakdown Part 1

3# Most of the functions follow a similar pattern where events array would be iterated through and for every plan_ID it would find all the workouts contained in the plans_workouts table with that id then either create an array of values such as total weight for each workout or return the most common workout name or category. Below shows one example of calculating a user's favourite workout. **(FR 10)**

```
//check if either array is empty
if(!empty($events) and !empty($WorkoutsArr)){
    //loop through events array
    for ($i=0; $i <count($events) ; $i++) {

        //loop through workouts array
        for ($x=0; $x <count($WorkoutsArr) ; $x++) {
            //find matching plan ID's
            if ($events[$i]["plan_ID"] == $WorkoutsArr[$x]["plan_ID"]) {


                //if the array key exists add 1;
                if (array_key_exists($WorkoutsArr[$x]["workout_name"],$favWorkoutArr)){
                    $favWorkoutArr[$WorkoutsArr[$x]["workout_name"]]++;
                }
                else{
                //if not create a new item with the key of the workout name and give it a value of 1;
                  $favWorkoutArr[$WorkoutsArr[$x]["workout_name"]] = 1;
                }

            }
        }
    }

    //Find the highest value in the array
    //by using PHP's max function.
    $maxVal = max($favWorkoutArr);

    //Use array_search to find the key that
```

1# Profile data pulled from the database using the session ID and a SQL Query.

```
$stmt = $con->prepare('SELECT user_email, user_pass FROM userprofile WHERE user_ID = ?');
```



2# The crucial part of these requirements was to deliver visually appealing and descriptive data regarding a user's workouts. This began by creating a number of functions contained in "statfunc.php" to calculate a user's category totals, workout totals, duration, weight, etc. I thought it necessary to give the user the ability to breakdown their statistics into 3 distinct time periods, all time, this month, and this calendar week. This involved creating 3 queries of event data at particular time periods, the results of these queries are then piped into the aforementioned functions. 10 Functions in total, ranging from getting the total number of plans and returning an array of workouts completed with that time period to calculating the

4# For the statistics regarding max unit (meaning duration or reps) and weight or in other words the users personal bests for each workout. This was calculated slightly differently, since I wanted the website to be fast and responsive it was necessary to place this data into a multidimensional associative array containing the name of the workout, it's ID, max unit, max weight, total unit, total weight and what unit was being used (reps or duration). This means that once passed to the JavaScript front end it is no longer necessary for the website to make a call to the server as all information is stored in local variables allowing for seamless changing between time period or workout with a click of a dropdown. **(FR 10)**

```
for ($i=0; $i <count($allWorkouts) ; $i++) {

    ${$allWorkouts[$i]["workout_name"]} = Array("workout_ID" => $allWorkouts[$i]["workout_ID"],
                                                "workout_name" => $allWorkouts[$i]["workout_name"],
                                                "totalUnit" => totalUnit($allWorkouts[$i]["workout_ID"],$WorkoutsArr,$events,$allWorkouts),
                                                "totalWeight" => totalWeight($allWorkouts[$i]["workout_ID"],$WorkoutsArr,$events),
                                                "maxUnit" => maxUnit($allWorkouts[$i]["workout_ID"],$WorkoutsArr,$events,$allWorkouts),
                                                "maxWeight" => maxWeight($allWorkouts[$i]["workout_ID"],$WorkoutsArr,$events),
                                                "unit" => getWorkoutUnit($allWorkouts, $allWorkouts[$i]["workout_ID"]));
```

5# This is the first implementation of the chart.js functionality, it takes the category array from each time period, as previously discussed, and outputs it as a doughnut graph. Following when the user selects a different time frame, the doughnut chart is destroyed and remade using the current array of data. **(FR 11 + 13)**

```
//display all/month/week stats
$( "#view" ).change(function() {
    var selection = selectChange();
    if (selection == "All") {

        title = "All";
        doughnutData = catArrAll;


    }else if(selection == "Month"){

        title = "Month";
        doughnutData = catArrMonth;


    }else if(selection == "Week"){
```

6# This is second instance of chart.js being used here it displays the number of goals per week, this can be toggled to show monthly workout plans completed too using the dropdown time frame box used in the previous section. The number of works per month/week is calculated upon opening the webpage and is then passed to JavaScript, the same functionality is used as in the category breakdown chart, whereby when a different timeframe is selected the chart is destroyed and remade using the correct data object. The object has a key named the week starting timestamp and a value of the number of plans completed that week. **(FR 11 + 13)**

7# Here a user can set their workout goals and the number of times this was met is checked and displayed. **(FR 10)**

```
for (var key in workoutsPerWeek) {
    if (workoutsPerWeek.hasOwnProperty(key)) {
        if (workoutsPerWeek[key] >= goalsArr[0]["weekly_goal"]) {
            succesfulWeekGoal++;
        };
    }
}
$('#weekGoal').val(goalsArr[0]["weekly_goal"]);
$('#metWeekGoals').val(succesfulWeekGoal);

for (var key in workoutsPerMonth) {
    if (workoutsPerMonth.hasOwnProperty(key)) {
        if (workoutsPerMonth[key] >= goalsArr[0]["monthly_goal"]) {
            succesfulMonthGoal++;
        };
    }
}
$('#monthGoal').val(goalsArr[0]["monthly_goal"]);
$('#metMonthGoals').val(succesfulMonthGoal);
```

This is achieved by simply looping through the monthly/weekly plans completed objects and comparing them to the goals set in the database.

user_ID	monthly_goal	weekly_goal
11	2	4
17	1	1

If the user has no goals set the script creates an entry for them with 1 as the goal for both. This is done through a simple insert query.

8# The final use of chart.js is a line chart used to track weight. Weight is stored as its own table in the database.

ID	user_ID	weight	entry_date
5	11	70	2021-03-25 11:49:56
6	11	71	2021-03-25 11:50:06
7	11	80	2021-03-25 11:53:15
8	11	77	2021-03-26 07:10:30
9	11	81	2021-03-26 11:04:12
10	17	11	2021-03-28 03:23:09
11	11	85	2021-03-30 02:28:37

The script makes a query to the database using the session ID, this then becomes the weight json object where the entry date is the key, and the value is the weight. This is provided as the data and x axis labels for the chart.

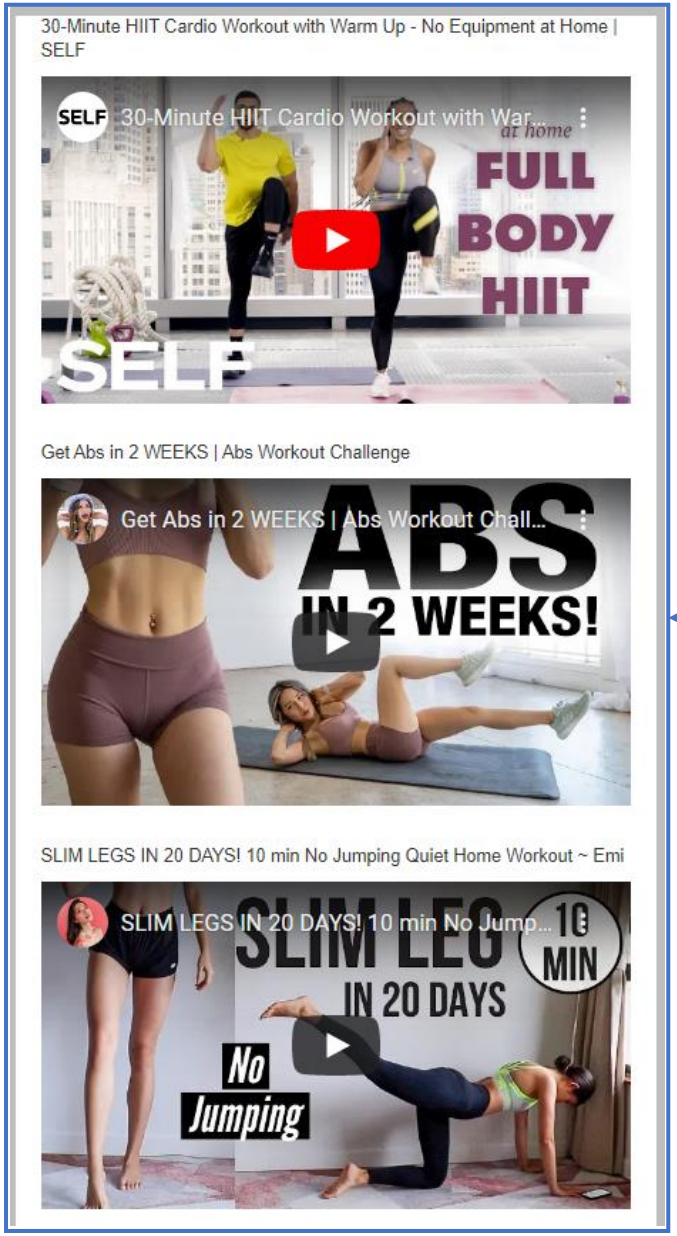
```
labels: labels,
datasets: [{
    label: 'Graph Line',
    data: data,
    backgroundColor: "#25CCF7",
    borderColor: "#FD7272",
    fill: false
}]
```

This enables users to track their weight over time **(FR 12 + 13)**

9# This final box in the profile page is where users can set their current weight, upon typing their weight in and hitting set it appends the data to the database using the "addWeight.php" file.

```
$sql = "INSERT INTO weight (user_ID, weight, entry_date) values ('$userID','$weight','$date');";
    //run query
mysqli_query($con,$sql);
```

Content Aggregation Breakdown



The tips and tricks page works off of the same statistics functions that were created in the previous section and are contained with the php file “statfunc.php”, these functions allow the webpage to find the favourite and least favourite, categories and workouts. Below is an example of how the favourite category is computed.

```php
function findFavCategory($catArr){
        //Find the highest value in the array
        //by using PHP's max function.
        //Use array_search to find the key that
        //the max value is associated with
    if (count($catArr)> 1) {
        $maxKey = array_search(max($catArr), $catArr);
        return $maxKey;
    }else{
        reset($catArr);
        $first_key = key($catArr);
        return $first_key;
    }
};
```

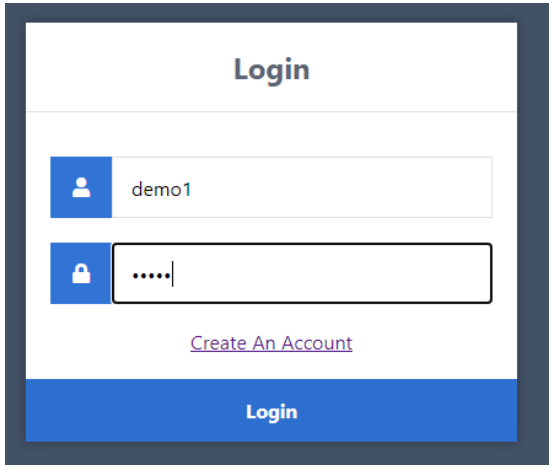
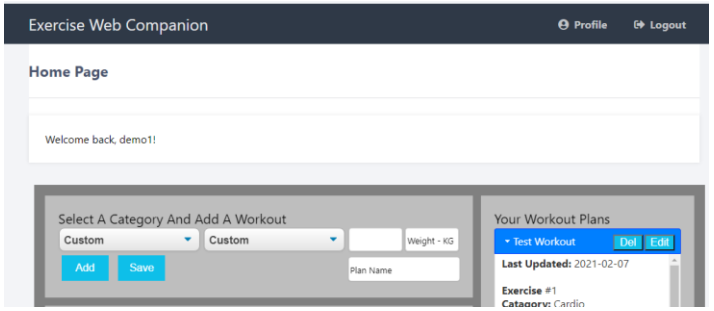
The category array is passed into the function where it is validated to ensure that it has more than 1 entry, next the highest value within the category array is used to find the respective key. The highest value indicates the number of times this category has appear in the users completed plans. This name of the category is then passed back to php and used as the value in a hidden html text box, this is how it is passed to the JavaScript portion of the page.

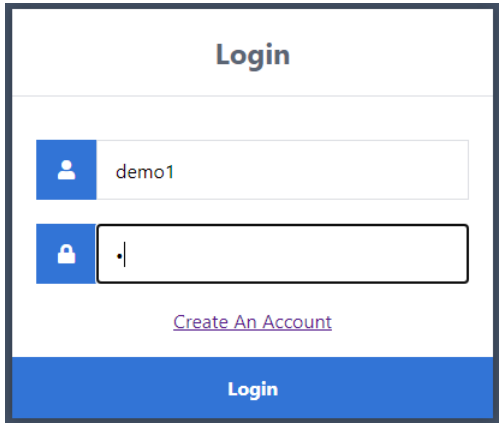
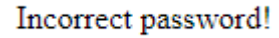
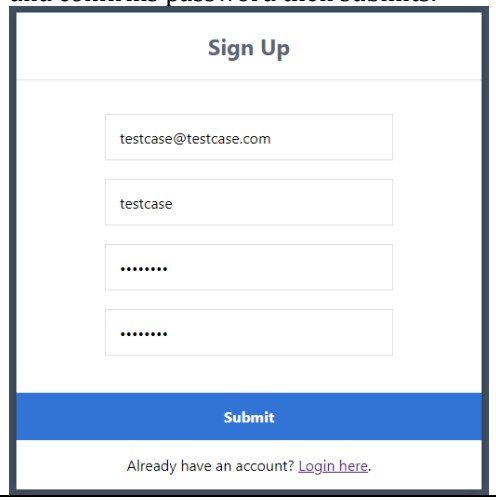
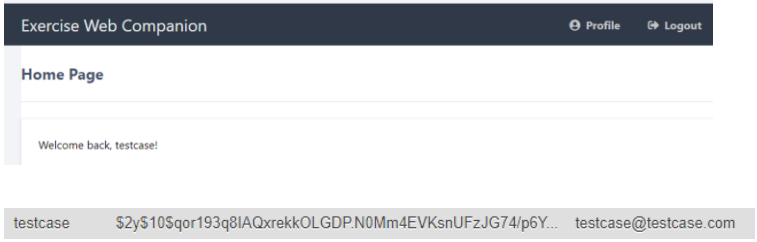
Once here we check if the value is none and if not concatenate it with the text “workout exercise” this is to ensure later that we limit our searches to exercise only. We place all these values into an array and pass it to the search function. This function pulls the most popular video by views and then appends it to the results div where it is embedded. **(FR 14)**

```javascript
function search(input) {
  var request = gapi.client.youtube.search.list({
    part: "snippet",
    type: "video",
    q: input,
    maxResults: 1,
    order: "viewCount",
    publishedAfter: "2015-01-01T00:00:00Z",
```

E. Appendix E - Verification

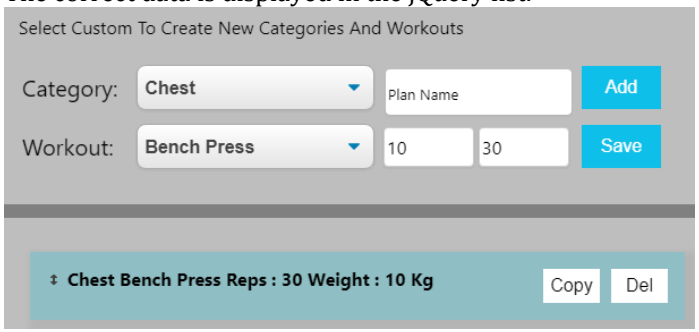
E.1. Create an Account Verification

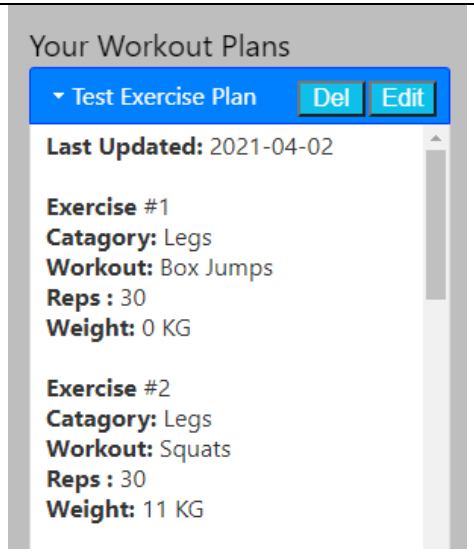
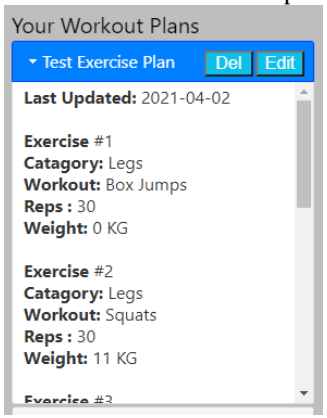
Create An Account Verification Table				
Requirement	Acceptance Criteria	Test + Screenshot	Expected Outcome	Outcome + Screenshot
Log in Securely (FR 1)	*Given:* That a user has a correct username and password *When:* They enter details and click login *Then:* They should proceed into the website opening a session and the password should be hashed.	The user inserts a correct password and username. 	The user is logged in and home.php is displayed.	As expect the home.php file is opened when a correct password and username is submitted. 

Log in Securely (FR 1)	*Given:* That a user has a correct username and password *When:* They enter details and click login *Then:* They should proceed into the website opening a session.	The user inserts an incorrect password and username. 	An error message is displayed.	As expect an error message is displayed when an incorrect password and username is submitted. 
Create an Account (FR 2)	*Given:* That a user has a correctly entered a new username, password, and Email *When:* They enter details and click create an account *Then:* A new account with those details should be created and a password is stored in the database as a hash.	The User inserts an email, username, password and confirms password then submits. 	Information is stored on the database and password as a hash then "testcase" is able to login.	"Testcase" is able to login and information is stored securely on database. 

E.2. Plan A Workout Verification

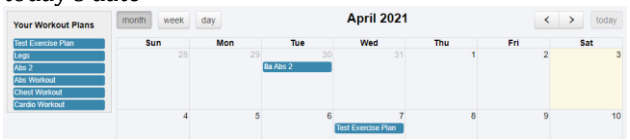
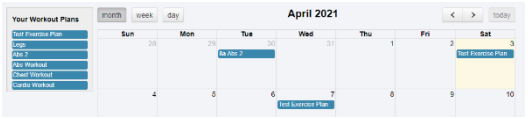
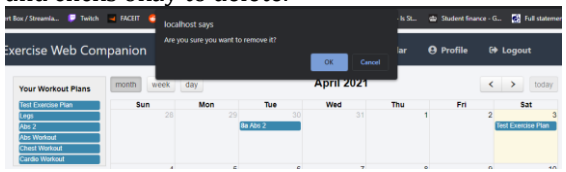
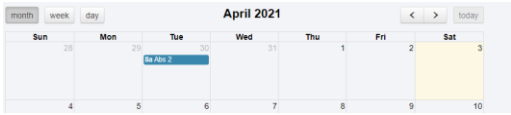
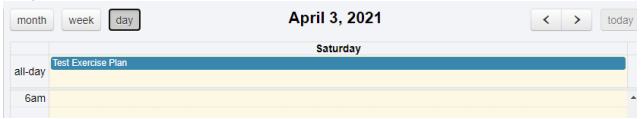
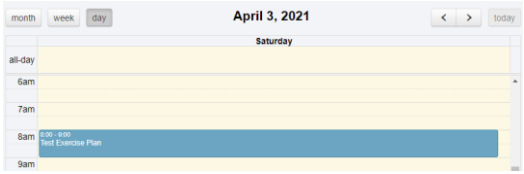
Plan A Workout Verification Table				
Requirement	Acceptance Criteria	Test + Screenshot	Expected Outcome	Outcome + Screenshot

Add Workout (FR 4)	Given: That a user is on the plan a workout page When: They have selected a category, workout, time/reps and weight and pressed "add" Then: A widget containing the given data should appear.	A weight, duration/reps is given and the Add button is pressed. 	The data is displayed in the jQuery list.	The correct data is displayed in the jQuery list. 
Sort Workouts (FR 3)	Given: That a user has workouts When: They drag a widget Then: It should move with the cursor and snap to the desired area once dragged.	In the workout list the user moves the fourth workout to the first. 	The fourth workout moves to being the to the first.	The fourth workout moves to being the to the first. 
Save Workout Plans (FR 5)	Given: A user has made a plan When: They click save	A workout is made, and the user hits save.	The workout Appears in the "Your Workout Plans" with the same data as entered.	The workout Appears in the "Your Workout Plans" with the same data as entered.

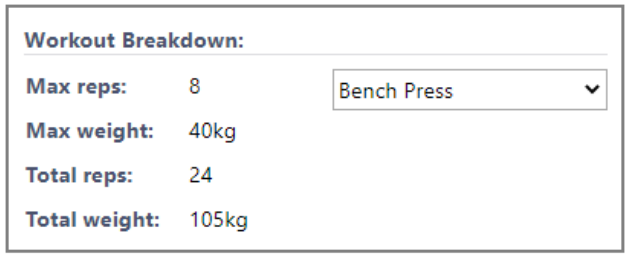
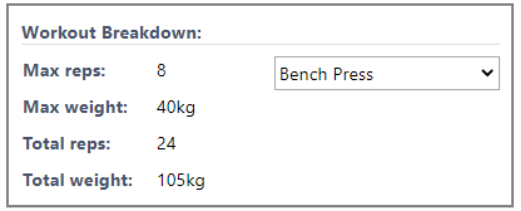
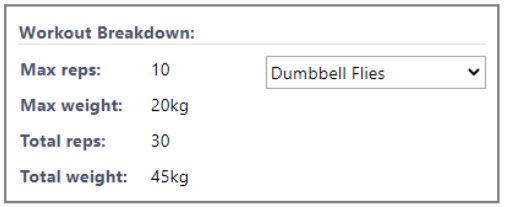
	Then: All data should be saved to the server.			
Retrieve and Edit Workout Plans (FR 6)	Given: A user has selected a plan. When: They click edit. Then: The workout should be displayed.	The user clicks edit on a plan. 	The selected workout appears with the same data as in the accordion.	The selected workout appears with the same data as in the accordion. 

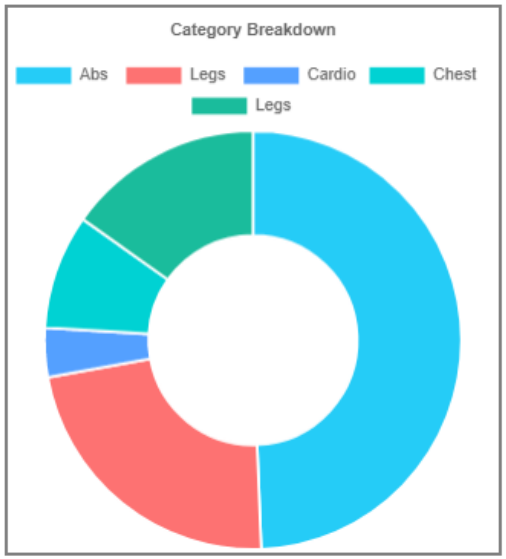
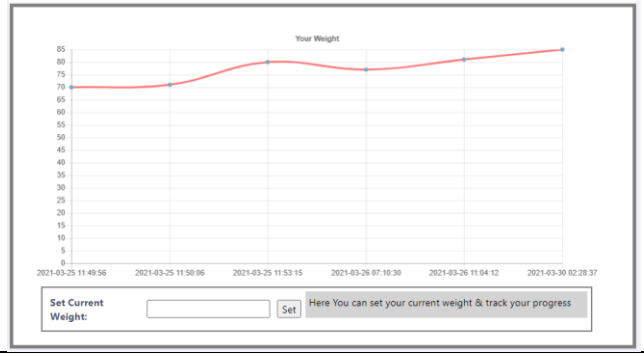
E.3. Schedule A Workout Verification

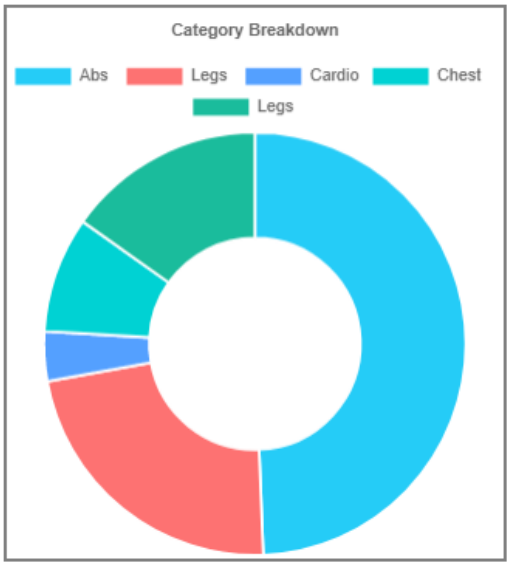
Schedule A Workout Verification Table				
Requirement	Acceptance Criteria	Test + Screenshot	Expected Outcome	Outcome + Screenshot

Schedule Workout Plans (FR 7)	Given: That the user has workout plans. When: The user drags a workout plan on to the calendar Then: The calendar should show where the plan was dragged too and save to the database	The user drags “test exercise plan” into the calendar on today’s date 	The Test exercise plan appears on today’s date and is saved in the database with allDay as true, and the correct time and date.	The Test exercise plan appears on today’s date.  The exercise plan is in the database with allDay as true, and the correct time and date. 
Delete Workout Plans (FR 8)	Given: That the user has workout plans on the calendar When: The user clicks and drags the event to another day or time. Then: The event should update to where it was dragged and update in the database	The user clicks on “test exercise plan” on todays date and clicks okay to delete. 	The Test exercise plan is removed from the calendar and database	The Test exercise plan is removed from the calendar.  The Test exercise plan is removed from the database.
Alter Workout Plans (FR 9)	Given: That the user has workout plans on the calendar When: The user clicks and drags the event to another day or time Then: The event should update to where it was dragged and update in the database	The user clicks on “test exercise plan” in day view then moves it into 8am and then drags the event to resize it to just 1 hour. 	The event starts at 8am and last for 1 hour on the calendar and the same in the database and appears as allDay 0.	The event starts at 8am and last for 1 hour on the calendar.  The event starts at 8am and last for 1 hour in the database and appears as allDay 0. 

E.4. Generating Statistics Verification

Generating Statistics Verification Table				
Requirement	Acceptance Criteria	Test + Screenshot	Expected Outcome	Outcome + Screenshot
Logging Personal Bests (FR 10)	Given: That the user has completed a workout at least once. When: The user views the statistics page. Then: The highest 1 rep max weight should be shown.	When the user opens the profile page, and selects a workout, it shows their max weight lifted. 	When the user opens the profile page, and selects a workout, it shows their max weight lifted.	When the user opens the profile page, and selects a workout, it shows their max weight lifted. 
Seeing Exercise Breakdown (FR 11)	Given: That the user has completed a workout at least once When: The user views the statistics page Then: The weight total and max are shown alongside the total and max reps or duration and a breakdown of the categories used.	When the user opens the profile page, and selects a workout, the weight total and max are shown alongside the total and max reps or duration and a breakdown of the categories used.	The correct total weight and max are shown alongside the total and max reps or duration are shown in the bottom widget and the doughnut chart displays a category breakdown.	The correct total weight and max are shown alongside the total and max reps or duration are shown in the bottom widget and the doughnut chart displays a category breakdown. 

				
Weight Tracker (FR 12)	Given: That the user has entered their weight at least once When: The user views the statistics page Then: The weight should be displayed on a chart showing progress.	When the user opens the page, it should open to a graph with 6 data points on, then should go and add 1 more with the value of 70. 	The graph should now show 7 data points with the additional value of 70.	The graph now shows 7 data point with the additional value of 70 added. 
Visual Aid's (FR 13)	Given: That the user has completed a workout at least once. When: The user views the statistics page. Then: The user should be presented with data, charts and graphs detailing their completed exercises.	Open the profile page.	All 3 graphs should appear and with the correct data.	All 3 graphs appeared with the correct data.

E.5. Content Aggregation Verification

Generating Statistics Verification Table				
Requirement	Acceptance Criteria	Test + Screenshot	Expected Outcome	Outcome + Screenshot

Tailed Personalised Tips (FR 14)	Given: That the user has an account When: The user clicks on the tips & tricks page. Then: The user should be presented with tailored YouTube content regarding their exercise habits.	Click on the tips and tricks page.	Embedded YouTube videos appear regarding exercise.	Embedded YouTube videos appear regarding exercise. 30-Minute HIIT Cardio Workout with Warm Up - No Equipment at Home | SELF  Get Abs in 2 WEEKS | Abs Workout Challenge  SLIM LEGS IN 20 DAYS! 10 min No Jumping Quiet Home Workout ~ Emi 

F. Appendix F – User Reviews

F.1. Create an Account User Reviews

User	Positives	Negatives.
1	The login page is easy to understand. Visually Appealing style.	There is no remember me button
2	Intuitive structure and design	I feel the password validation isn't secure enough
3	Placeholder text and images help to make it easy to understand what to do	none
4	Neat design makes logical sense.	Password hint or password recovery
5	Nice styling	none

F.2. Plan a Workout User Reviews

User	Positives	Negatives.
1	I like how the placeholder text changes depending on the exercise. The last updated feature is nice to see when this exercise was edited.	Some functionality is not clear due to poor labelling or font choice.
2	I like how you can create unique custom exercise for yourself.	I don't like how it resets my plan if I create a new workout or category
3	I like how you can rearrange and sort exercises in your plan. I like the consistency of theme and colour throughout.	Some of the functionality isn't clear immediately requires reading so it's intuitive.
4	The workout plans section is easy to read and understand. The colour choice makes it very easy to read for someone who is colour-blind.	Before adding workouts, the middle of the page is empty.
5	I like that I can name my plans the same thing and they don't interfere with each other. The drop-down boxes are easy to use and clear.	The workout plans list thing isn't very clear how to operate it.

F.3. Schedule a Workout User Reviews

User	Positives	Negatives.
1	Today's date stands out well. It's easy to schedule a workout.	Text size is quite small.
2	It's very intuitive, and easy to understand how to use without any instruction.	Only shows A for Am and P for Pm, this could be confusing.
3	I like the consistency of theme and colour throughout.	Large empty spaces on the left.
4	I like how my exercises appear automatically on the left, this makes it very easy for me plan my routines.	Sometimes if the title is too long it hangs over the edge of the blue rectangle.
5	I like how the calendar is continuous. I like the minimalistic design.	none

F.4. Generating Statistics User Reviews

User	Positives	Negatives.
1	The category breakdown is really visually appealing, useful and would be motivating. Switching between the time frames is very easy and is a good idea. Very valuable. The weight line chart is really easy to understand, and I can imagine it would be easy to track progress.	The goals met section sort of blends in with the other sections, I feel like there should be more emphasis on this part.
2	The category breakdown is helpful to guide me in having a balanced exercise routine. Very Comprehensive statistics.	Maybe there should be a button to change your username and email.
3	Interesting statistics and could be a good source of motivation. I like the bright choice of colours on the graphs.	I don't like the emphasis on body weight, I feel like this could build unhealthy habits and thought processes.
4	It's easy to see the breakdown of workouts in one place due to the dropdown.	It's not clear from the menu bar above that this page would contain stats, maybe there could be some form of hint within other pages.

	I like how the graphs respond to the cursor by showing the dates and values of each data part.	
5	The monthly/ weekly goals are good for positive reinforcement, I like how you can personalise them and it shows how many times you've met it. Very cool. The text boxes are informative and helpful.	I really like this page, no negatives.

F.5. Content Aggregation User Reviews

User	**Positives**	**Negatives.**
1	I like how the YouTube videos embed into the page. I like how the title of the video is above them.	I think the videos could be a little bigger, the menu title for this page doesn't really explain what it is.
2	This would really help in find new exercise to do. Apparently, these videos are tailored to me personally, this is a really interesting feature in my opinion.	It's not obvious that these videos are unique to me, maybe this needs to be made clearer.
3	I like that I can stay on the website and watch these videos. It is also nice that you can open and view multiple videos at a time.	The web page styling is sort of bland and boring, it doesn't match with the other pages I've seen. This is off putting
4	I like that the videos don't auto play. I like that I can go full screen on the videos.	The styling isn't great, seems very boring and lazy
5	It's really cool that these videos are tailored to me and are just one click away. This makes the website feel like a one stop shop for all my exercise needs.	Not a fan of the styling and I feel like there needs to be more of an explanation of what's going on in this page.

G. Appendix G – User Stories

G.1. Create an Account User Stories

Log in Securely (FR 1)	**Must Have**	**3 Days**
As a user of this application I want to securely login. So that I can use the application from my profile without threat of attackers.		
Acceptance Criteria: *Given:* That a user has a correct username and password *When:* They enter details and click login *Then:* They should proceed into the website opening a session.		

Create an Account (FR 2)	**Must Have**	**3 Days**
As a new user of this application I want to create a new account. So that I have a personalised experience		
Acceptance Criteria: *Given:* That a user has a correctly entered a new username, password, and Email *When:* They enter details and click create an account *Then:* A new account with those details should be created and a password is stored in the database as a hash		

G.2. Plan a Workout User Stories

Add Workouts (FR 3)	**Must Have**	**14 Days**
As a person that wants to workout I want to have a library of exercises that I can select from So that I can easily plan a workout.		
Acceptance Criteria: *Given:* That a user is on the plan a workout page *When:* They have selected a category, workout, time/reps and weight and clicked “add” *Then:* A widget containing the given data should appear		

Sort Workouts (FR 4)	**Must Have**	**14 Days**
As a person that finds organised workout plans useful to helping my motivation I want to be able to easily sort my exercises into the order I desire. So that I can quickly form a workout plan to my taste.		
Acceptance Criteria: *Given:* That a user has workouts *When:* They drag a widget *Then:* It should move with the cursor and snap to the desired area once dragged.		

Save Workout Plans (FR 5)	**Must Have**	**3 Days**
As a person that regularly exercises I want to quickly create and save plans. So that I can later come back to them.		

Acceptance Criteria: *Given:* A user has planned *When:* They click save *Then:* All data should be saved to the database

G.3.Schedule a Workout User Stories

Retrieve and Edit Workout Plans (FR 6)	**Must Have**	**3 Days**
As a person that regularly exercises I want to be able to retrieve old plans. So that I can stick to a workout routine and build good habits.		
Acceptance Criteria: *Given:* A user has selected a plan. *When:* They click edit. *Then:* The workout should be displayed.		

Schedule Workout Plans (FR 7)	**Must Have**	**4 Days**
As a person that regularly exercises I want to be able to schedule a workout plan. So that I can organise my routine.		
Acceptance Criteria: *Given:* That the user has workout plans *When:* The user Drags a workout plan on to the calendar *Then:* The calendar should show where the plan was dragged to and save to the database		

Delete Workout Plans (FR 8)	**Must Have**	**2 Days**
As a person that regularly exercises I want to be able to delete workout plans from my schedule. So that I can organise my routine		
Acceptance Criteria: *Given:* That the user has workout plans on the calendar. *When:* The user clicks on an event. *Then:* The event should be deleted from the database and the calendar.		

Alter Workout Plans (FR 9)	**Must Have**	**2 Days**
As a person that regularly exercises I want to be able to alter plans in my schedule. So that I can organise my routine and stay motivated		
Acceptance Criteria: *Given:* That the user has workout plans on the calendar *When:* The user clicks and drags the event to another day or time *Then:* The event should update to where it was dragged and update in the database		

G.4. Generating Statistics User Stories

Logging Personal Bests (FR 10)	**Must Have**	**2 Days**
As a person that regularly exercises I want to be able to log my personal bests and weight. So that I can track my progress.		
Acceptance Criteria: *Given:* That the user has completed a workout at least once *When:* The user views the statistics page *Then:* The highest 1 rep max weight should be shown.		

Seeing Exercise Breakdown (FR 11)	**Must Have**	**2 Days**
As a person that regularly exercise I want to be able to view a comprehensive breakdown of my max weight, reps and total weight and reps. So that I can track my progress		
Acceptance Criteria: *Given:* That the user has completed a workout at least once. *When:* The user views the statistics page. *Then:* The weight total and max is shown alongside the total and max reps or duration and a breakdown of the categories used.		

Weight Tracker (FR 12)	**Must Have**	**2 Days**
As a person that wants to stay in shape I want to be able to track my personal weight and how I've progressed. So that I can track my progress		
Acceptance Criteria: *Given:* That the user has entered their weight at least once *When:* The user views the statistics page *Then:* The weight should be displayed on a chart showing progress		

Visual Aid's (FR 13)	**Must Have**	**2 Days**
As a person that wants to have a balanced workout routine I want to be able to see graphs, charts and data detailing my exercise. So that I stay motivated and see my weak areas.		
Acceptance Criteria: *Given:* That the user has completed a workout at least once *When:* The user views the statistics page *Then:* The user should be presented with data, charts and graphs detailing their completed exercises.		

G.5.Content Aggregation User Stories

Tailed Personalised Tips (FR 14)	**Must Have**	**2 Days**
As a new person to exercise I want to be able to have easy to access tips and tutorials. So that I learn how to exercise effectively.		
Acceptance Criteria: *Given:* That the user has an account *When:* The user clicks on the tips & tricks page. *Then:* The user should be presented with tailored YouTube content regarding their exercise habits.		