

Gym Analytics and Personal Training Application
Final Report



Author: Jamie Hawthorne

Supervisor: Greg Doyle

Submission Date: 30/04/2021

Abstract

The purpose of this “Gym Analytics and Personal Training application” is to develop a mobile application for Android and iOS with two sections available to the public. Firstly, we have a client section to help the average person hire a personal trainer, purchase a workout plan whether it is a pre-made plan or a requested plan. The client also has the option to set motivational personal goals/ milestones for them to achieve.

Secondly, we have the trainer side of this application, it is here where a Personal Trainer can grow their business by posting pre-made plans, create custom plans for clients, reach a wide range of potential clients and receive payments directly into their PayPal accounts.

Table of Contents

Abstract.....	2
Table of Figures	5
Introduction.....	6
Project Description.....	7
Application Screenshots.....	8
Splash Screen	8
Login Screen	8
Reset Password Screen	9
Registration Screen	9
Client’s Initial Home Screen.....	10
Client’s Home Screen completed.....	10
Client’s Goals Screen.....	11
Client’s Add Goals Screen.....	11
Client’s Manage Goal Screen	12
Client’s View Trainers Screen	12
Client’s Selected a Trainer Screen	13
Client’s View Plans Screen.....	13
Client’s Selected a Plan Screen.....	14
Client’s Request Custom Plan Screen.....	14
Client’s Payment Screen	15
Client’s Daily Workout Completed Screen	15
Client’s Daily Workout Completed Screen	16
Change Password Screen	16
Close Account Screen	17
Image Options Screen	17
Settings Screen.....	18
Trainer’s Home Screen	18
Trainer’s Payments Screen	19
Trainer’s View Clients Screen	19
Trainer’s View Plans Screen.....	20
Trainer’s View Selected Plan Screen.....	20
Trainer’s Add Plans Screen	21
Trainer’s Edit Plan Screen	21
Trainer’s View Requests Screen	22

Trainer’s View Selected Request Screen	22
Admin Home Screen	23
Admin Settings Screen.....	23
Admin Manage Clients Screen	24
Admin Manage Trainers Screen	24
Admin Add Exercises Screen	25
Admin Send Reset Link Screen	25
Application Dashboard	26
Issues Encountered.....	27
iOS	27
Checkout system	27
Storing the Workout Plans	27
Finding an Exercise Dataset/API	28
Adherence to Specification and Design	28
Changes to the Design	28
Welcome Screen	28
View Client’s Profiles.....	29
Trainer Payment Transactions	29
Assigning Workouts.....	29
Learning Outcomes	29
Technical.....	29
Xamarin.....	29
Firebase	30
Stripe payments.....	30
PayPal Payouts.....	30
Personal.....	31
Time Management	31
Developing a Plan Structuring	31
Project Review	31
Project Success.....	31
Project Failures	32
Future Development.....	33
Conclusion	33
Acknowledgments.....	33
Declaration.....	34

Table of Figures

Figure 1. Splash Screen.....	8
Figure 2. Login Screen.....	8
Figure 3. Reset Password Screen	9
Figure 4. Registration Screen.....	9
Figure 5. Client’s Initial Home Screen	10
Figure 6. Client’s Home Screen completed	10
Figure 7. Client’s Goals Screen	11
Figure 8. Client’s Add Goals Screen	11
Figure 9. Client’s Manage Goals Screen	12
Figure 10. Client’s View Trainers Screen.....	12
Figure 11. Client’s Selected a Trainer Screen	13
Figure 12. Client’s View Plans Screen	13
Figure 13. Client’s Selected a Plan Screen	14
Figure 14. Client’s Request Custom Plan Screen	14
Figure 15. Client’s Payment Screen.....	15
Figure 16. Client’s Daily Workout Screen	15
Figure 17. Client’s Daily Workout Completed Screen	16
Figure 18. Change Password Screen.....	16
Figure 19. Close Account Screen.....	17
Figure 20. Image Options Screen.....	17
Figure 21. Settings Screen	18
Figure 22. Trainer’s Home Screen.....	18
Figure 23. Trainer Payments Screen	19
Figure 24. Trainer’s View Client’s Screen	19
Figure 25. Trainer’s View Plans Screen	20
Figure 26. Trainer’s View Selected Plan Screen	20
Figure 27. Trainer’s Add Plan Screen.....	21
Figure 28. Trainer’s edit Plan Screen	21
Figure 29. Trainer’s View Requests	22
Figure 30. Trainer’s View Selected Request	22
Figure 31. Admin Home Screen	23
Figure 32. Admin Settings Screen	23
Figure 33. Admin Manage Clients Screen.....	24
Figure 34. Admin Manage Trainers Screen.....	24
Figure 35. Admin Add Exercises Screen.....	25
Figure 36. Admin Send Reset Link Screen.....	25
Figure 37. Firebase Analytics	26

Introduction

The purpose of this final report is to document the development process of this “Gym Analytics and Personal Training” application as well as a retrospective and reflection on the development process from start to finish.

The first section of this document is focused on looking at the final submission that will be presented along with a summary on how each page works. This section will be broken down further into three subsections:

1. Application Screenshots and Description
2. Application Dashboard
3. Issues Encountered

The second section of this document compares the final submission to the functional specification and design manuals respectively explaining all changes that have occurred throughout the development process. This section will also contrast the initial prototype designs to the final product explain the need for the changes that occur.

The third section covered in this document will be the learning outcomes that have been achieved through developing this application. This section will include the technical skills gained, such as implementing firebase authentication into an application, along with personal skills and learnings, such as the importance of time management.

The fourth section of this document will present an overall review of the completed application. This section will include the projects successes, failures, what I would do differently given the opportunity and the potential for future development within this application.

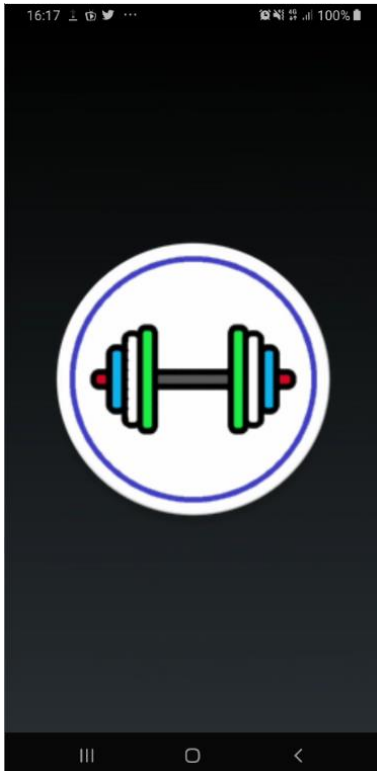
Project Description

This “Gym Analytics and Personal Training” application was built as a cross platform application using Xamarin as the development platform. Xamarin enables developers to share an average of 90% of their application across platforms. It does this by allowing the developer to develop all the functional code in a single language C#. The User Interface (UI) is then developed using Microsoft’s Extensible Application Markup Language (XAML) allowing the application to be developed as a cross platform application while acting as if it was a native application. Xamarin’s development platform was run inside Microsoft’s Visual Studio giving this application access to an extensive library of addons and SDK’s in the form of NuGet packages.

The backend of this application was developed using Googles Firebase BaaS (Backend as a Software) platform. The database section of the application is handled by Firebase Realtime Database which is an in cloud-hosted NoSQL database that allows the application to store and sync data in Realtime. This application also takes advantage of other Firebase features such as Firebase Authentication which securely registers user by Salt and Hashing the passwords along with storing them in a separate location from the login information only accessible by the application. Another feature of Firebase used by the administrator of this application is Firebase Analytics. Firebase Analytics processes raw information from the application in real time providing the administrator with a vast amount of knowledge from the most used version to what city the application is most popular in.

Application Screenshots

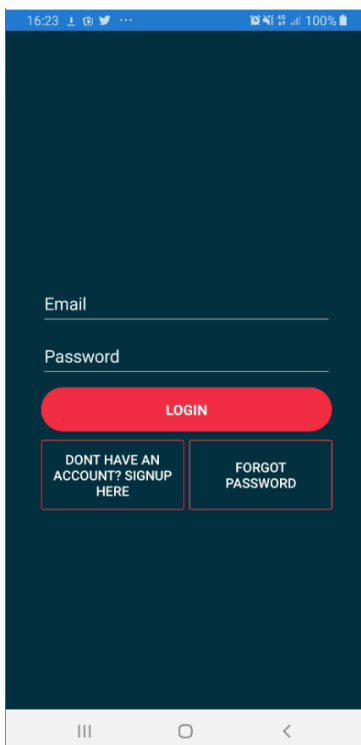
Splash Screen



This is the application's Splash screen which appears every time the application is launched.

Figure 1. Splash Screen

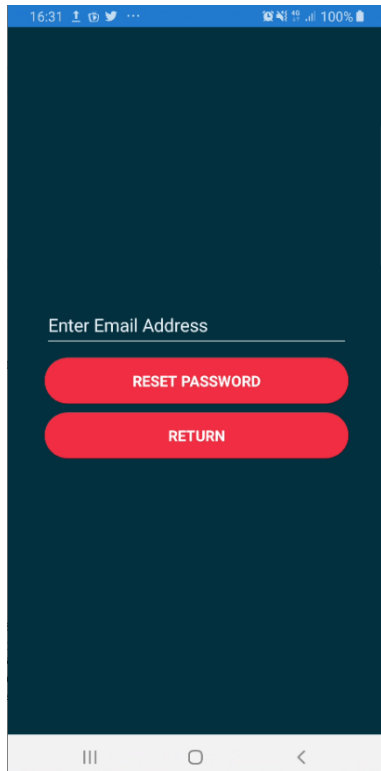
Login Screen



This is the application's Login Screen for every user. Here the user can either enter their email and password or if they have forgotten their password, navigate to the forgot password page. If the user does not yet have an account, they can also navigate to the registration page from this screen.

Figure 2. Login Screen

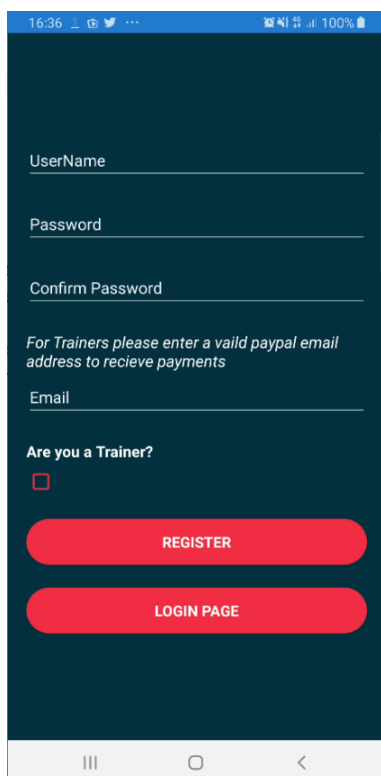
Reset Password Screen



This is the application's Reset Password Screen. Here the user can enter their email address and request a password reset link to be sent to them from Firebase directly.

Figure 3. Reset Password Screen

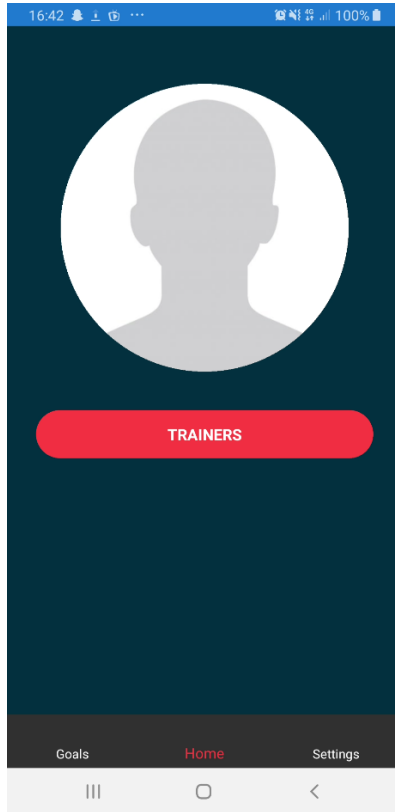
Registration Screen



This is the application's Registration Screen for every user. Here the user enters a username, password, confirmation of password, email address and finally is asked to tick if they are a trainer. If the user ticks the box, they are registered as a trainer within the database otherwise they are registered as a client.

Figure 4. Registration Screen

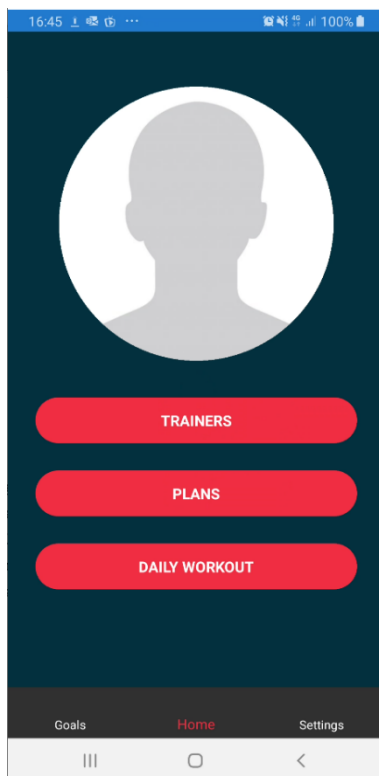
Client's Initial Home Screen



This is a Client's Home Screen if they have just registered with the application and have not yet selected a trainer or purchased a plan. From here the client can currently access "View Trainers Screen" or use the tabbed navigation bar at the bottom to navigate to the "Goals" or "Settings" Screens.

Figure 5. Client's Initial Home Screen

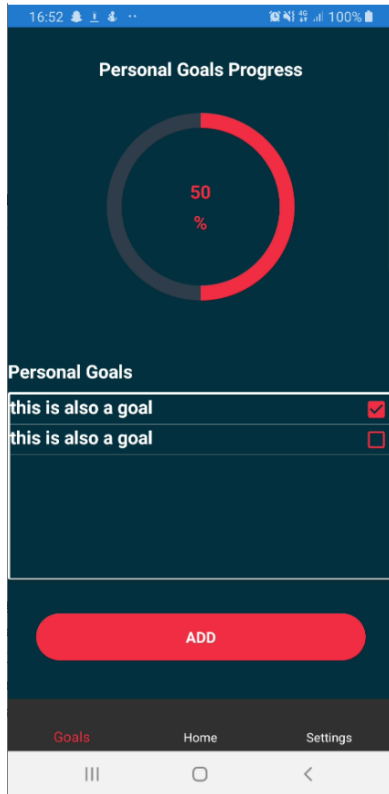
Client's Home Screen completed



Here we can see the Client's Home Screen completed and the client can now access the Daily Workout and Plans sections as they have selected a trainer and purchased a plan.

Figure 6. Client's Home Screen completed

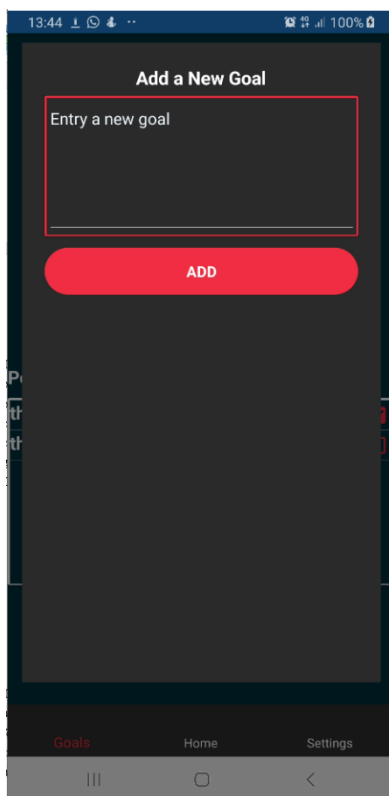
Client's Goals Screen



This is the Client's Goals Screen. Here the client can set personal goals that they wish to accomplish and check their progress at any time. The user simply taps the goal (not the check box) and is displayed with the option to mark the goal as completed or simply remove the goal entirely. The client adds goals by clicking the add button and navigating to the Add Goals screen.

Figure 7. Client's Goals Screen

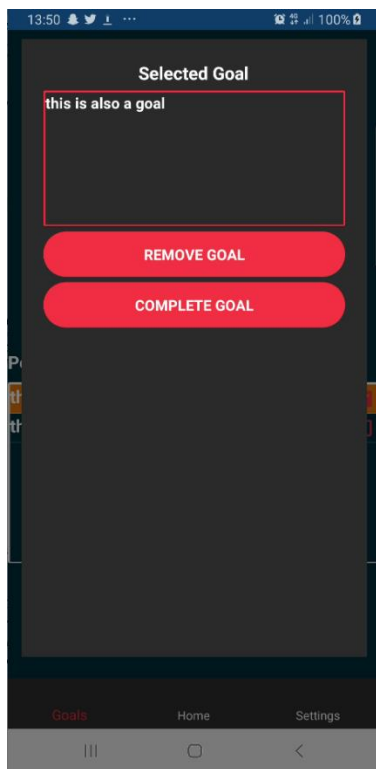
Client's Add Goals Screen



This is the Client's Add Goals Screen. Here the client enters personal goals that they wish to accomplish and then clicks the "Add" button to add them to their personalised list.

Figure 8. Client's Add Goals Screen

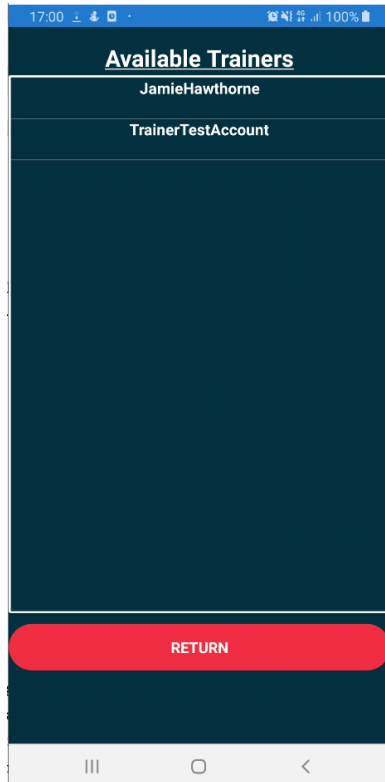
Client's Manage Goal Screen



This is the Client's Manage Goals Screen. Here the client has the option to mark the selected goal as either done using the "Completed" button/The client may wish to remove the goal entirely from the list using the "Remove" button.

Figure 9. Client's Manage Goals Screen

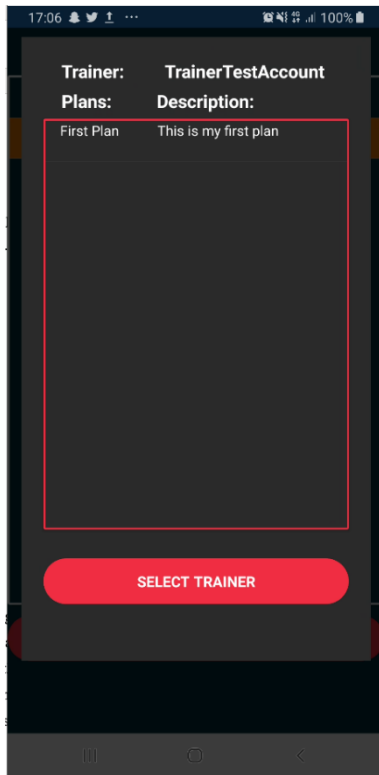
Client's View Trainers Screen



This is the Client's View Trainers Screen. Here the client can view all registered trainers in the application. The client simply needs to tap on a trainer's name to get a list of their pre-made plans to get an idea of what the trainers specialise in.

Figure 10. Client's View Trainers Screen

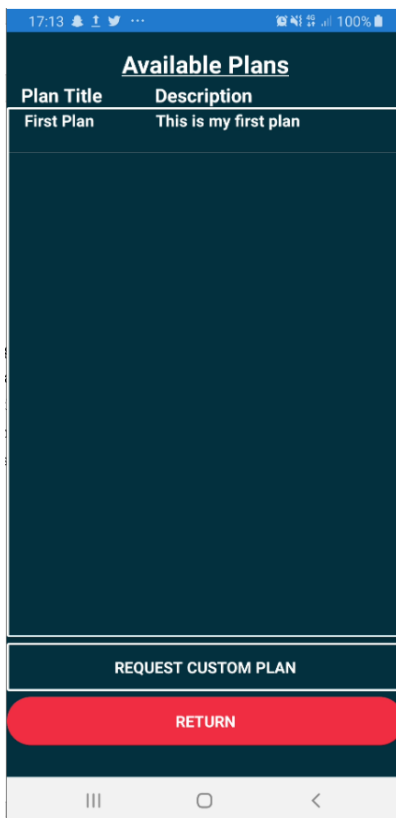
Client's Selected a Trainer Screen



This is the Client's Selected a Trainers Screen. Here the client can view all the trainer's pre-made plans and a brief description of the plan. If the user wishes to select this trainer as their personal trainer, they simply press the "select trainer" button.

Figure 11. Client's Selected a Trainer Screen

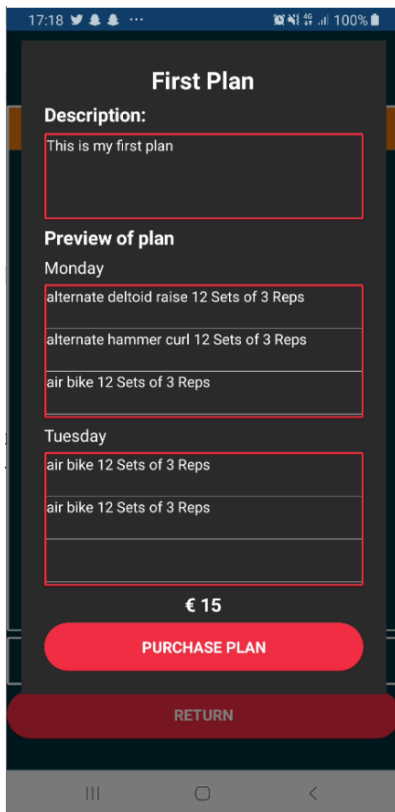
Client's View Plans Screen



This is the Client's View Plans Screen. Here the client can view all the trainer's pre-made plans and select one for more information and pricing. The client also has the option to request a custom plan to suit their needs.

Figure 12. Client's View Plans Screen

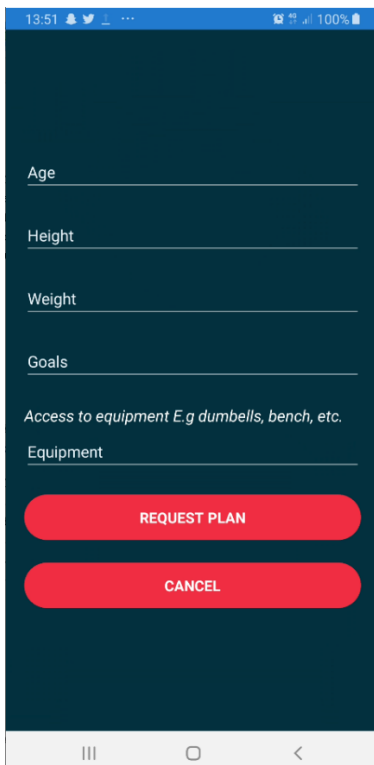
Client's Selected a Plan Screen



This is the Client's Selected a Plan Screen. Here the client has selected a plan to view more information about the plan. The client gets a short preview of the plan along with a description and price. If the client likes the plan, they can choose to purchase it now using the "Purchase Plan" button.

Figure 13. Client's Selected a Plan Screen

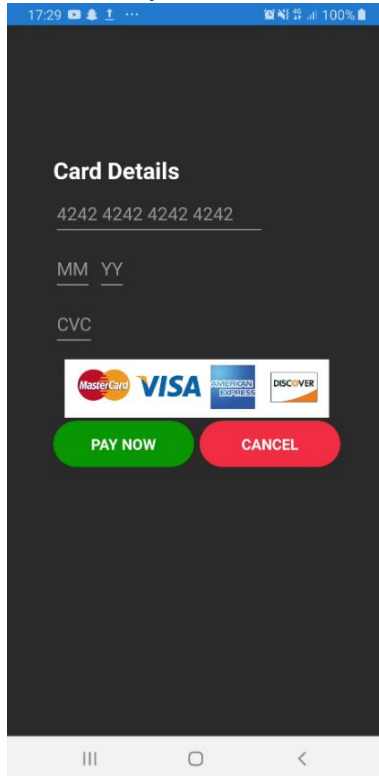
Client's Request Custom Plan Screen



This is the Client's Request a Custom Plan Screen. Here the client is asked to enter more information about themselves so the trainer can create a plan to suit them. To request a plan the client simply fills out the form (not all fields are required although they are recommended for the best results) and then click the "Request Plan" button.

Figure 14. Client's Request Custom Plan Screen

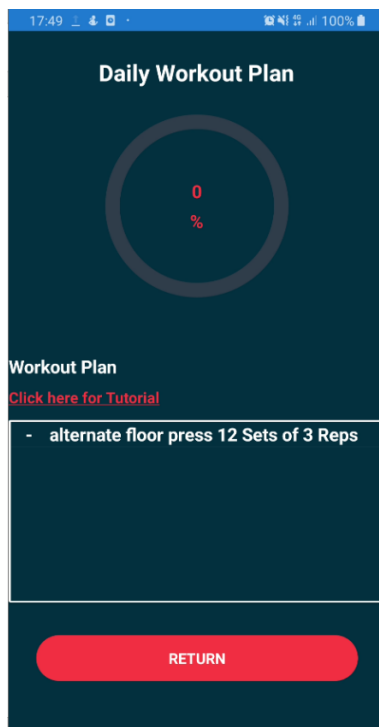
Client's Payment Screen



This is the Client's Payment Screen. Here the client is asked to enter their credit card details to pay for the custom or pre-made plan. The client has the option to cancel at this time before any charges are made.

Figure 15. Client's Payment Screen

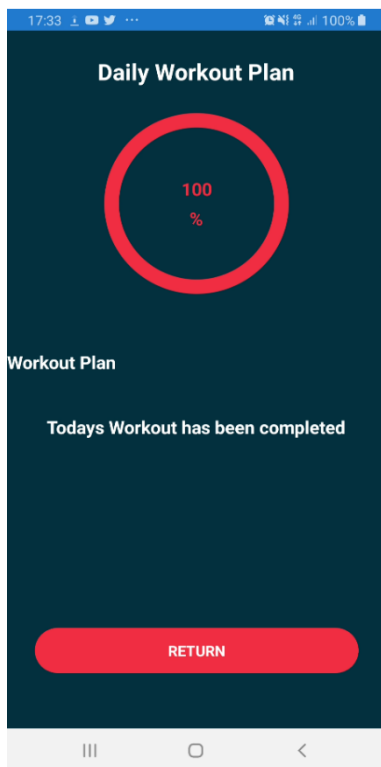
Client's Daily Workout Completed Screen



This is the Client's Daily Workout Screen. Here the client is displayed with a progress bar to let them know how far along in the workout they are. The client also has the option to click the tutorial link if the trainer has provided one (this could be a YouTube tutorial or a website of theirs showing the client how to complete the exercise). Once the client has completed the exercise, they simply tap the exercise and it is removed from the list and the progress bar is updated.

Figure 16. Client's Daily Workout Screen

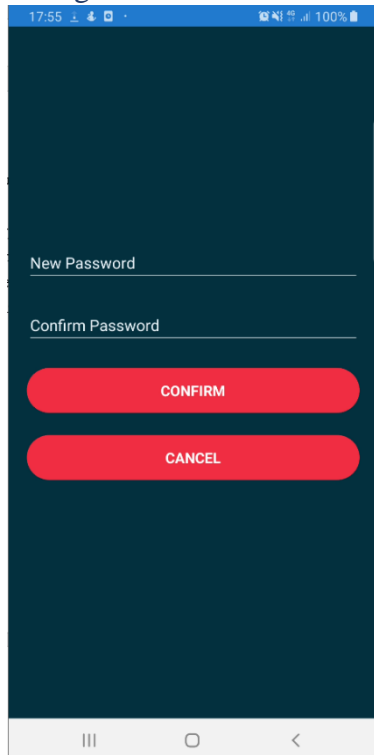
Client's Daily Workout Completed Screen



This is the Clients Daily Workout Completed Screen. Here the client has been informed they have completed their workout for the day.

Figure 17. Client's Daily Workout Completed Screen

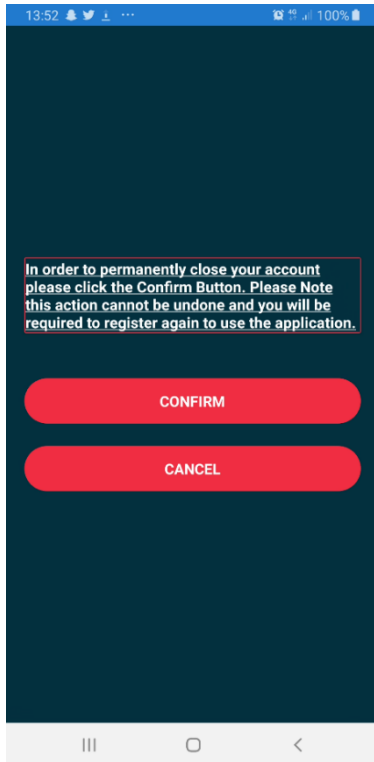
Change Password Screen



This is the application's Change Password Screen for all users registered with the app. The user simply inputs a new password and then types the password again to confirm it is correct. The user then clicks the "Confirm" button to update the password. The user also has the option to navigate back to the home screen by using the "Cancel" button.

Figure 18. Change Password Screen

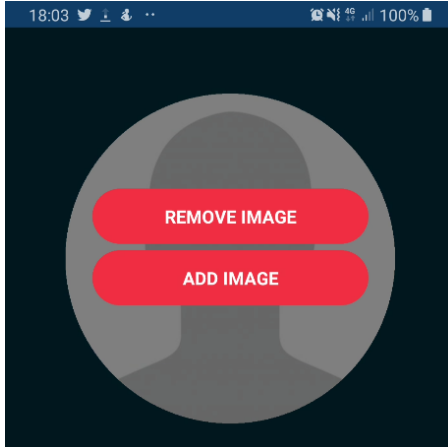
Close Account Screen



This is the application's Close Account Screen for all users registered with the app. The user simply clicks the "Confirm" button to close their account permanently. The user also has the option to navigate back to the home screen by using the "Cancel" button.

Figure 19. Close Account Screen

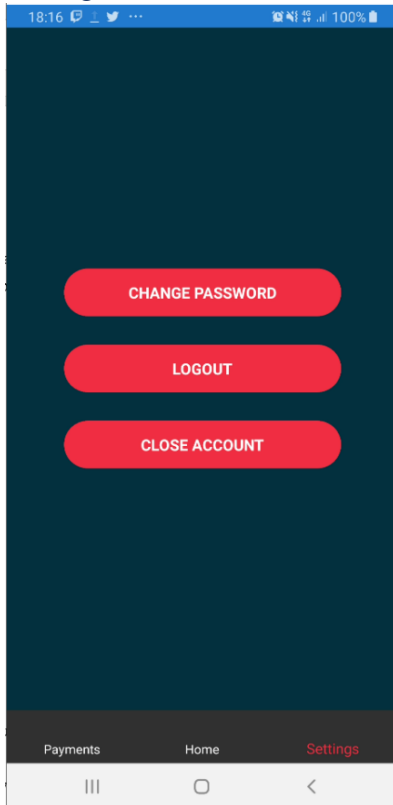
Image Options Screen



This is the application's Image Options Screen for all users registered with the app. The user simply taps on the profile picture to display these options. The user is then able to remove their uploaded image or add an image. The user can also choose to do neither and tap anywhere to close these options. If the user chooses to add an image, they are then given the option to upload from gallery or take a new image with their camera.

Figure 20. Image Options Screen

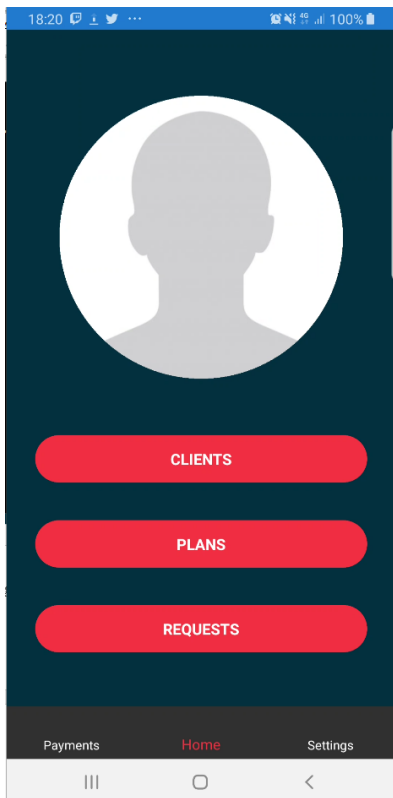
Settings Screen



This is the application's Settings Screen for all users registered with the app. From here the user can change their password, logout and even close their account.

Figure 21. Settings Screen

Trainer's Home Screen



This is a Trainer's Home Screen. From here the trainer can access the "View Clients", "View Plans", "View Requests" Screens or use the bottom tabbed navigation to access the "Payments" or "Settings" Screens

Figure 22. Trainer's Home Screen

Trainer's Payments Screen



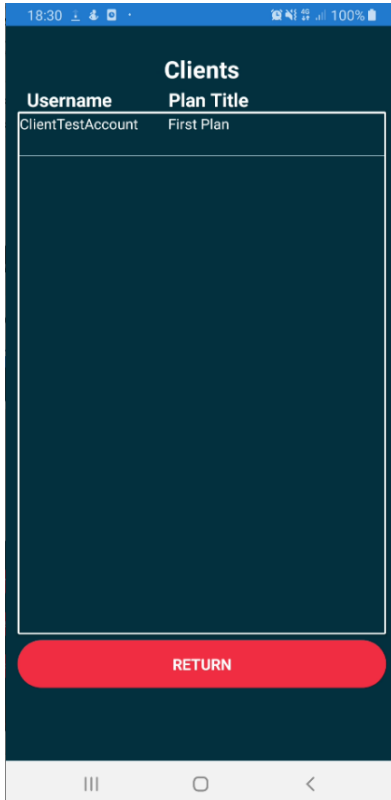
The screenshot shows a mobile application interface for a Trainer's Payments Screen. At the top, the status bar displays the time 18:26, signal strength, Wi-Fi, and 100% battery. Below the status bar is a table with four columns: ClientName, PlanTitle, Date, and Amount. The table contains ten rows of data, all with the same values: ClientName: ClientTestAccount, PlanTitle: First Plan, Date: 29/04/2021, and Amount: € 15. At the bottom of the screen, there is a navigation bar with three items: Payments (highlighted in red), Home, and Settings. Below the navigation bar is a standard Android navigation bar with three icons: a hamburger menu, a home button, and a back arrow.

ClientName	PlanTitle	Date	Amount
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15
ClientTestAccount	First Plan	29/04/2021	€ 15

This is a Trainer's Payments Screen. Here the trainer can view all payments received for the period set by the administrator (a month). The trainer can choose to sort the data using any of the different headings and by either ascending or descending order.

Figure 23. Trainer Payments Screen

Trainer's View Clients Screen



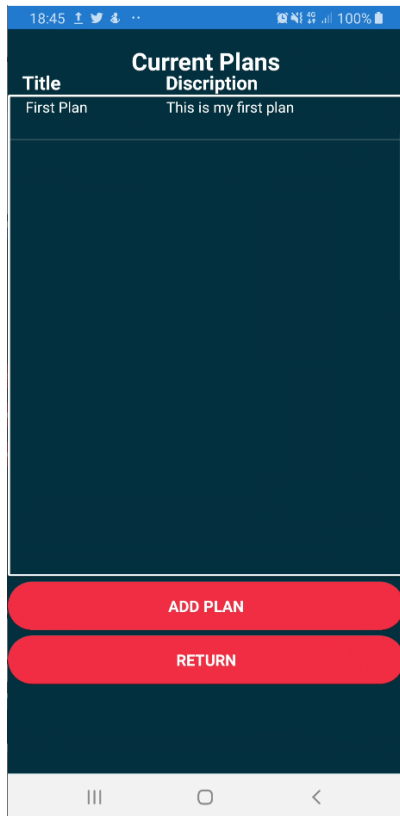
The screenshot shows a mobile application interface for a Trainer's View Clients Screen. At the top, the status bar displays the time 18:30, signal strength, Wi-Fi, and 100% battery. Below the status bar is a dark blue header with the title "Clients". Underneath the header is a table with two columns: Username and Plan Title. The table contains one row of data: Username: ClientTestAccount, Plan Title: First Plan. Below the table is a large red button with the text "RETURN". At the bottom of the screen, there is a navigation bar with three items: a hamburger menu, a home button, and a back arrow.

Username	Plan Title
ClientTestAccount	First Plan

This is a Trainer's View Clients Screen. From here the trainer can view all clients they are currently selected by and what plans those clients are currently using.

Figure 24. Trainer's View Client's Screen

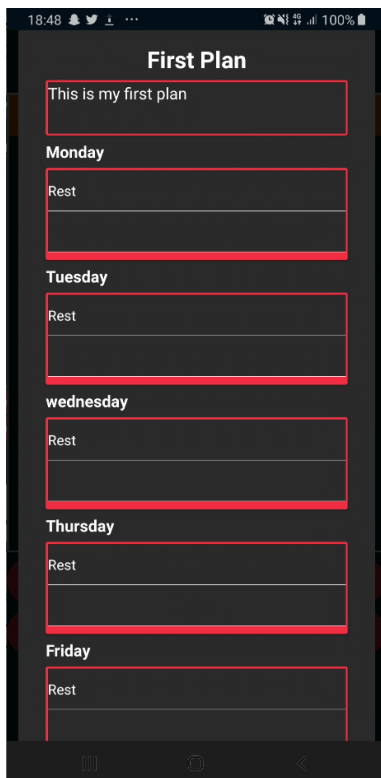
Trainer's View Plans Screen



This is a Trainer's View Plans Screen. Here the trainer can view all the pre-made plans they have made. The trainer can either choose to select a plan and view the entire plan, Click the "Add Plan" button to create a new plan or click the "Return" button to navigate back to the home screen.

Figure 25. Trainer's View Plans Screen

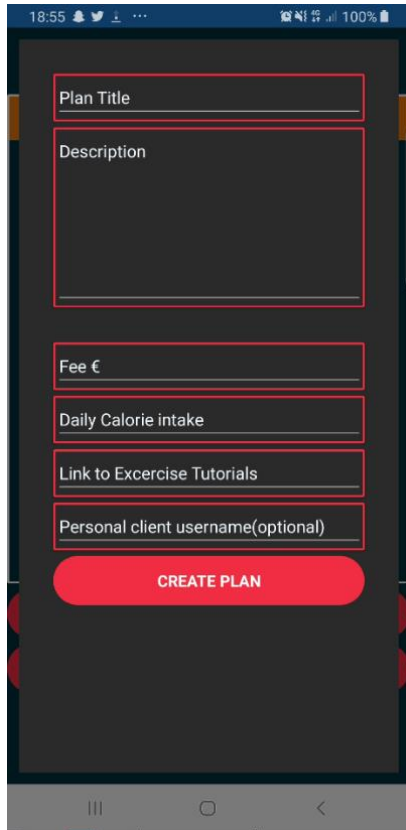
Trainer's View Selected Plan Screen



This is a Trainer's View Selected Plan Screen. Here the trainer can view the all the details of the plan they have selected including the tutorial link, cost, if it was made for a specific user and every day's workout routine. The user is also presented with the options to either edit or delete the plan.

Figure 26. Trainer's View Selected Plan Screen

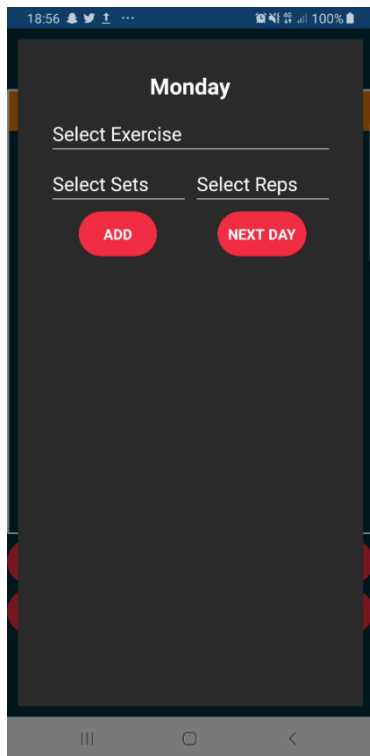
Trainer's Add Plans Screen



This is the Trainer's Add Plan Screen. Here the trainer creates a plan by selecting an exercise, the number of sets and reps for this exercise. The trainer then selects the "Add" button to assign that to a given day. Once the trainer has Added all the exercises for a day, they select next day and repeat the process. If the user wishes to set a "rest" day, they simply press next day without adding an exercise to the current day. Once all the days have been completed the user enters the remaining plan details which are the daily calorie intake, if it is for a specific client (so only they can see it), the plan title, description, the link to a tutorial and finally the fee.

Figure 27. Trainer's Add Plan Screen

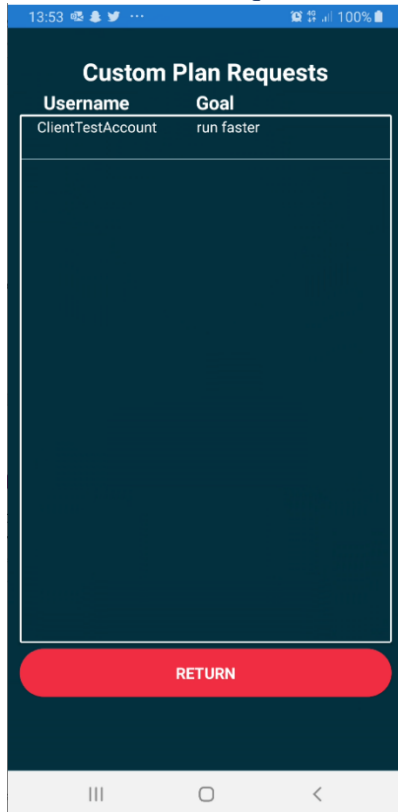
Trainer's Edit Plan Screen



The Trainer's Edit Plan Screen works identically to the Add plan screen however it is prepopulated with the end information.

Figure 28. Trainer's edit Plan Screen

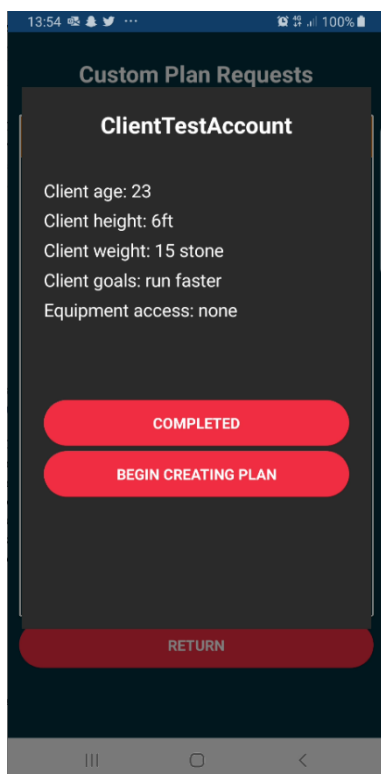
Trainer's View Requests Screen



This is a Trainer's View Request's Screen. Here the trainer can view all the requests made by clients for plans. The trainer can either choose to select a request and view the entire request or click the "Return" button to navigate back to the home screen.

Figure 29. Trainer's View Requests

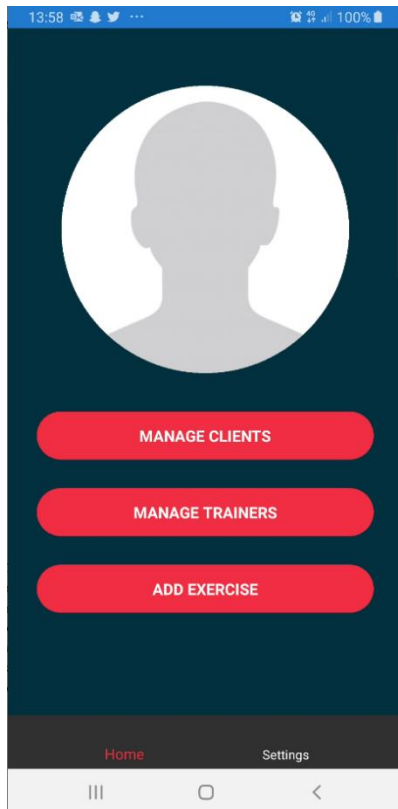
Trainer's View Selected Request Screen



This is a Trainer's View Selected Request Screen. Here the trainer can view all the information relation to the request. The trainer can mark the request as completed once the plan has been made or navigate directly to the "Add New Plan" page using the buttons on screen. The trainer can also navigate back to the "Trainer View Requests" page by clicking outside of the popup page.

Figure 30. Trainer's View Selected Request

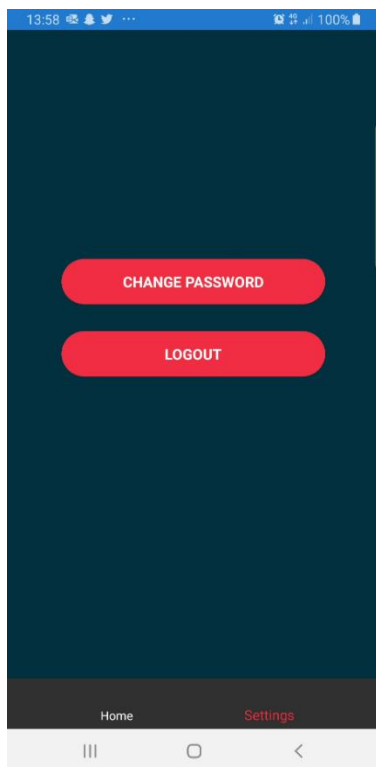
Admin Home Screen



This is the Admins Home Screen. Here the admin can view complete lists of trainers and clients within the application. The trainer also has the option to add new exercises to the database or navigate to the “Settings” using the bottom navigation panel.

Figure 31. Admin Home Screen

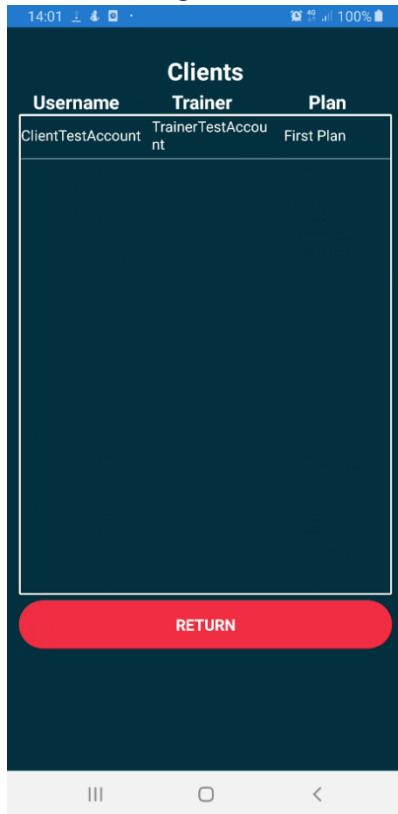
Admin Settings Screen



This is the Admins Settings Screen. Here the admin can change their password or logout. The only difference between this screen and the general users’ “Settings” screen is the admin cannot close their account.

Figure 32. Admin Settings Screen

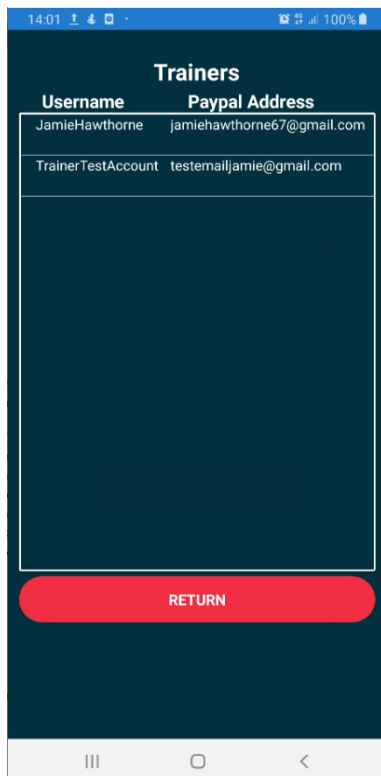
Admin Manage Clients Screen



This is the Admins Manage Clients Screen. Here the admin can view a full list of all registered clients within the application. The admin can then select a client to be given the option to send a password recovery link.

Figure 33. Admin Manage Clients Screen

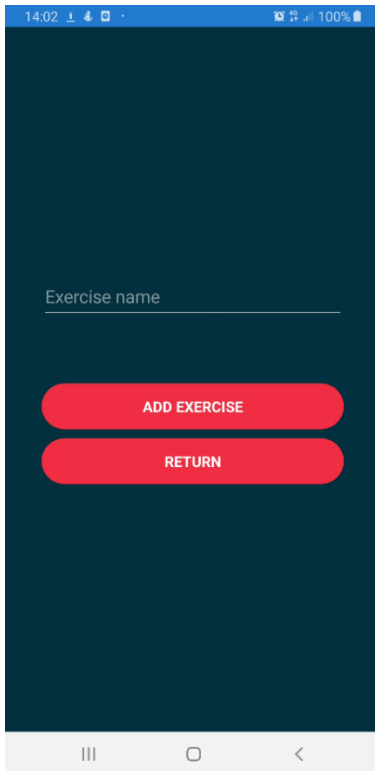
Admin Manage Trainers Screen



This is the Admins Manage Trainers Screen. Here the admin can view a full list of all registered trainers within the application. The admin can then select a trainer to be given the option to send a password recovery link.

Figure 34. Admin Manage Trainers Screen

Admin Add Exercises Screen



This is the Admins Add Exercises Screen. Here the admin can directly add new exercises to the database for the trainers to choose from.

Figure 35. Admin Add Exercises Screen

Admin Send Reset Link Screen



This is the Admins Send Reset Link Screen. Here the admin can send a password recovery link to the selected user. In future developments the admin may have more features such as disable account implemented here however it is currently not supported.

Figure 36. Admin Send Reset Link Screen

Application Dashboard

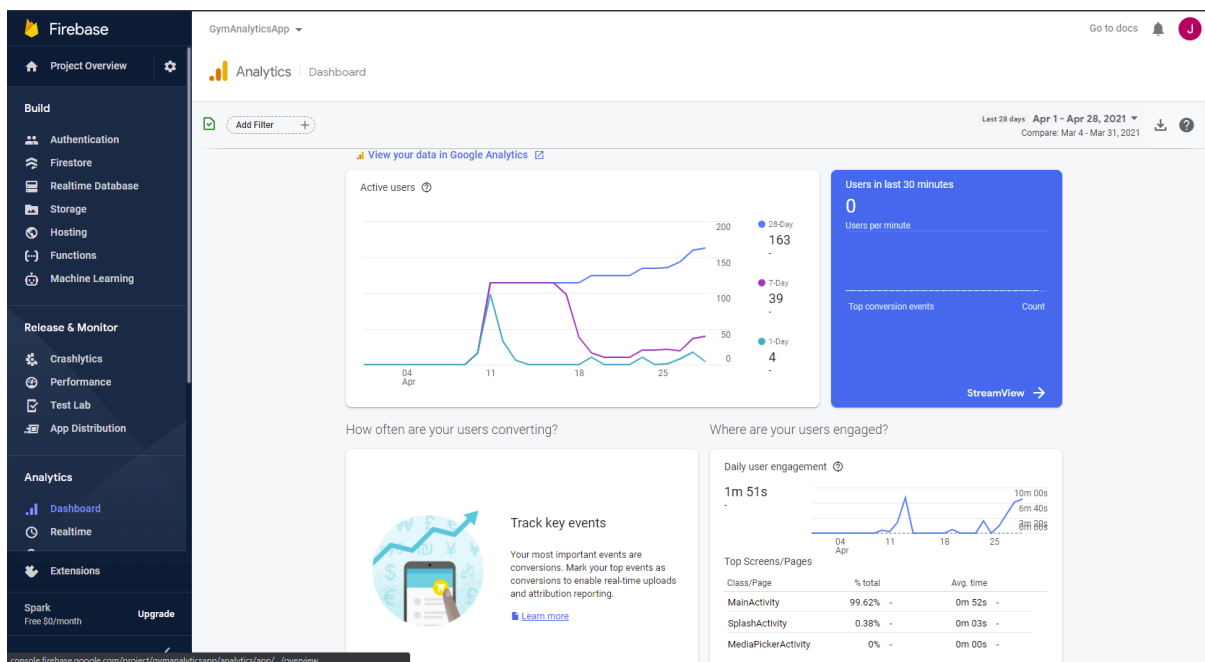


Figure 37. Firebase Analytics

Firebase Analytics acts as the application's dashboard for the administrator showing all data in real time. The benefit to having this dashboard means at any time the administrator can know exactly how many users are using the app, what version they are on and what sections of the app do not get used as much and perhaps need work. While in Firebase the administrator also can temporarily disable accounts as warnings, reset passwords or even close accounts completely. The original specification for this application was to include a suitable web dashboard to accompany this application and I feel Firebase Analytics does this and so much more.

Issues Encountered

iOS

One of the biggest issues I encountered while developing this application was the need for a mac device in order to fully develop the iOS side of this application. Originally when I undertook the task of developing a cross platform application, I saw many ways to emulate the iOS software in a virtual environment in order to still develop the Apple side of this application. However, when development began it became evident that none of these emulators worked as promised leading me to be unable to test and fully develop the iOS side of my application.

Checkout system

Initially when I was planning the development of this application, I intended to use a PayPal checkout system that would allow the client to sign into their PayPal account and pay the trainers directly. However, PayPal no longer support the payment from one standard PayPal account to another leading me to have to implement a middleman software to get the payments to the trainers. This could have been possible by simply having the clients pay the application's business PayPal and then in turn have the application's PayPal pay the trainers but then I got to thinking what if the clients do not have a PayPal account? This then led me to implement Stripe so that the clients can pay by card and in turn the trainers will be paid via PayPal from the applications account.

As the trainer receives money from the applications business account and this application is currently using PayPal Sandbox the only trainer registered who can receive money while the application is in the sandbox is:

Email: testemailjamie@gmail.com.

Password: Password12

Storing the Workout Plans

Prior to developing this application, I had no knowledge of Firebase or its Realtime Database, nor had I any previous experience with node-based storage. So, when it came to figuring out how to store the multiple exercises that would come together to form a workout, I was stumped. Originally, I had planned to store the workouts as an array with multiple strings contained inside them however, I was unable to find a way to do this within Firebase. This took me weeks to figure out, but I then came to the idea to have the exercise as one big string

and every time an exercise and its corresponding sets and rep counts are added I would also attach a “,” to the end. This then meant when I was returning the workout, I could parse the string into a list to be displayed back to the user.

Finding an Exercise Dataset/API

While developing this application I needed a dataset of exercises that exists for the trainer to choose from while they are creating a plan. This seemed like a simple request however after searching high and low I was unable to find a suitable API. The only one I could find is “updated” daily however it is also offline during this period which did not seem so bad until I realised it was offline for 2 hours a day in the middle of the day. After being unable to find an API, I then decided to build a dataset inside the database that the application can call upon. The only issue here is I still needed a list to implement into the database. After many days of searching, I found a GitHub containing an open-source list of certified exercises to implement into my database.

Adherence to Specification and Design

Throughout the development of this application the design has changed however the core functionality has stayed true. The main functionalities of this application laid out in the Functional Specification for clients was to allow them to register, select a personal trainer and purchase a plan. The main functionalities for a trainer were to allow them to CRUD plans and receive payments within the application. Each user section of this application also included subsections such as allowing the clients to set themselves personal goals and allowing the trainer to view all payments received.

Changes to the Design

Welcome Screen

Originally this application was designed with the intention for all users to have separate login screens and the initial screen to be a user selection screen. On this screen the user would also be given the options to register or reset their passwords. This however upon further review was seen as an unnecessary screen and the functionality has since been merged with the login screen which now also acts as the welcome screen.

View Client's Profiles

This initial plan for this application design had intended for the user's profiles to be visible by other user types, for example a given client could view a trainer's profile and vice versa. This was removed however due to further research as I discovered not all people using gym applications feel comfortable with this. Therefore, to appeal to more people this function and screen was removed.

Trainer Payment Transactions

Initially the trainer's payment screen was designed to show both "pending" and "received" transactions. However, while developing the application, it was designed in such a way that there is no pending transaction as payment is now required up front. A pending transaction or upcoming transaction screen may be integrated if a subscription system is later implemented into the application.

Assigning Workouts

I had initially intended to allow a trainer to assign a workout to a given client once the client requested a plan. However, this was removed to ensure the trainer develops a custom plan to suit the client as opposed to just assigning a plan they have previously made because it is similar to what the client needs.

Learning Outcomes

Technical

Throughout the development of this application, I was exposed to new technologies and programming languages that I either had limited, or no previous experience with. This led to me having to expand my knowledge base to complete this project. The new technologies are as follows:

Xamarin

Xamarin was used as my development platform to allow me to develop this application as a cross platform app. Prior to developing this application I had no previous experience with Xamarin, so it was all very new to me and required a lot of research.

Xamarin uses a language called XAML for all the front-end UI design and data binding, prior to working on this project I had never heard of XAML. However, with hours invested into research and a lot of trial-and-error I have successfully developed my entire UI in the language and can add it to my list of known languages.

Firestore

Prior to developing this application all my previous interactions with a database have been done through SQL. Firestore Realtime Database however is a cloud-hosted NoSQL database so I had no idea how to communicate with it. Again, through the vast amount of documentation provided by Firestore and a lot of trial-and-error testing I successfully implemented my “FirestoreAssistant” file which handles all interactions with my database.

Although the initial processes of learning to communicate with Firestore was a long and tedious experience, I am very glad I decided to incorporate it into my application. It is a service I will certainly use in future developments.

Stripe payments

While developing this application I required a way to receive payments from clients. Through a lot of research I discovered Stripe which is a service that I feel I have only scratched the surface of. Stripe allows a business to implement a payment system that will take in multiple forms of payments. This service is something I had no previous knowledge of, nor had I ever needed to implement a payment system before. With the help of the many sources of information provided by Stripe I was successfully able to incorporate a main feature of my app. Stripe is a fantastic service I am glad I discovered and will certainly implement if I am to require a payment system in further builds.

PayPal Payouts

The PayPal Payouts system was probably the greatest challenge of this application to implement. This was not because it was difficult to understand but due to the lack of publicly available information. While researching this service for the initial design and research stages of this application, PayPal seemed like the obvious choice for paying the trainers. However, when it came to finding sources to explain the implementation, PayPal’s documentation is severely lacking and took an extremely long time to find out how to incorporate this into my application. This is a technology I am unsure if I would use again in the future, however I am glad I have learned how to implement it on the off chance it is required.

Personal

Time Management

This project was the largest individual undertaking I have taken on during my time in college. Time management played a huge role in the development of this application because it was such a large project and because the application and accompanying materials had to be developed alongside my other modules. With the unique year we have had, the whole course was converted to online learning and many of the modules were converted to full continuous assessment. This meant we had to sit more exams throughout the year and needed to allocate time to study for these exams along with completing this project. These factors combined meant that time management was hugely important.

Developing a Plan Structuring

Prior to undertaking this project, I severely lacked the ability to develop and follow a structured plan, nor did I see the major importance of this skill. However, this became evident to me early in the development of this project as I was constantly moving back and forth between the two sides of my application which led to very little work being done. Once I developed a structured plan and began following it properly, I began to see progress very quickly. This is a skill I will certainly implement into every project I undertake in the future.

Project Review

The development of this project has been one of the most technically and personally challenging projects I have undertaken during my time in college. This is due to the overall size of the project and the new technologies I had never researched or implemented before. Overall, I feel this “Gym Analytics and Personal Training” application was successful when compared to the original requirements, design, and specification. The core functionality has successfully been implemented and the design has only changed slightly throughout.

Project Success

As the project currently stands, I completed most of functionality I had set out in the Functional Specification document under the “Metrics” heading. Those metrics are as follows:

- The application allows users to successfully register within the application.
- The application successfully accepts two categories of users and both categories can log in successfully.

- Users can smoothly and successfully navigate throughout the application. This feature was tested by a select few classmates I asked to test the application.
- All data required by the user is displayed in a simple and understandable manner throughout the application.
- Clients can successfully select trainers as their personal trainer.
- Clients can successfully CRUD personal goals for themselves.
- I successfully implemented a working payments system allowing the clients to purchase plans from the trainers.
- Trainers can successfully view a list of all their registered clients.
- Trainers can successfully CRUD plans for clients specifically or to be pre-made.
- The trainer is successfully able to track all payments received including the date it was received, the amount, what plan it was for and who paid it.

Additionally, I feel the application was also a success in these areas not listed in the original metrics:

- The admin is successfully able to monitor all analytically data collected from the application with the use of Firebase Analytics.
- I successfully built a certified list of exercises and implemented them into the application.
- The application has been successfully launched and is fully functional on Android.

Project Failures

As the project currently stands, the project has only failed under the following task in the “Metrics” heading. That metric is as follows:

- The application has failed to be developed for and run-on iOS.

This failure is due to the lack of equipment to test and complete the development for the iOS section of this application.

Future Development

Currently, I have no intention to develop this application further than its current state. This being said, I have outlined a few areas below I would look to develop further given the intention and time.

- I would acquire an apple mac product and complete the process of developing the application for Apple.
- Before release I would build and implement a Machine Learning system that would be used to verify a trainer's authenticity before their account is activated.
- I would continue to research and attempt to find an API updated regularly for the exercises. If this were unobtainable, I would attempt to build this API.
- I would hire a UI designer to make the application more aesthetically pleasing where it is currently lacking.

Conclusion

In conclusion this project was a fantastic challenge as a fourth-year project as it involved much research and implementation of new technologies. I am really pleased with the final product and despite not being able to test it in an Apple environment feel it was a success.

The finalised application code and APK can be found on my GitHub at:

<https://github.com/Jamie-Source/GymApp>

Acknowledgments

Firstly, I would like to say a big thank you to my project supervisor, Greg Doyle. Without his support and feedback throughout the year I don't feel I would have achieved the current standard of this project.

I would also like to take this opportunity to say a special thanks to Jack, Amy, and Thomas whose support has helped throughout the college year.

Declaration

- I declare that all material in this submission, e.g., thesis/essay/project/assignment, is entirely my own work except where duly acknowledged.
- I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams, or other material; including software and other electronic media in which intellectual property rights may reside.
- I have provided a complete bibliography of all works and sources used in the preparation of this submission.
- I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offense.

Student Name: Jamie Hawthorne

Student Number: C00226160

Student Signature: *Jamie Hawthorne*

Date: 30/04/2021