Instiúid Teicneolaíochta Cheatharlach

INSTITUTE *of* TECHNOLOGY CARLOW

At the Heart of South Leinster

Final Year Project, Design Manual, Gym and Analytical App/Dashboard.

Author: Kevin Quinn.

Student No: C00216607.

Supervisor: Greg Doyle.

Table of Contents

1. Introduction

The reasoning for creating this document is to describe how this application, Revolute Fitness will function. This document continues from the functional specification document, which described the functionality of the system. This document will outline and explain the design of the overall architecture of this system, a description of the cloud infrastructure and a design of user interfaces. The databases design will also be listed to show how a user's data will be stored and accessed.
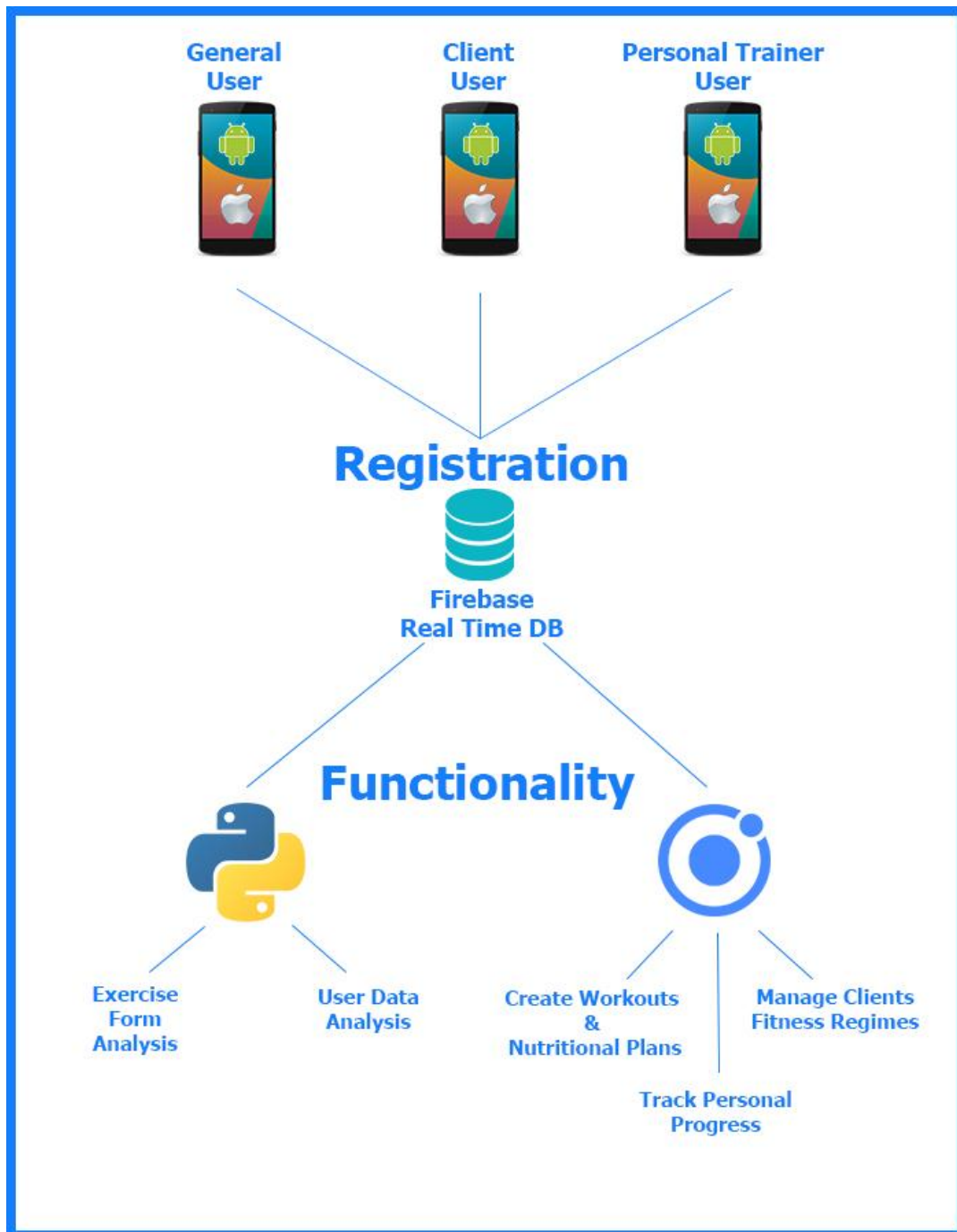
With this document, any developer should be able to create or work on this applications system as it will contain all the architecture and design plans needed to develop the application.
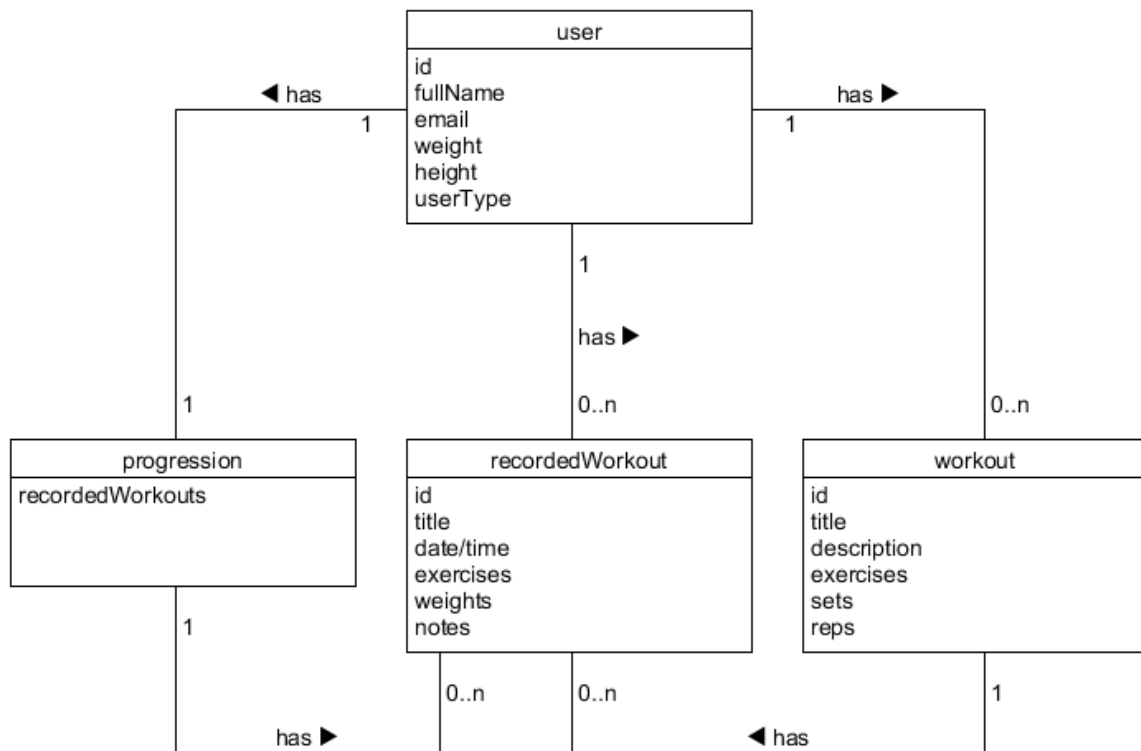
2. High Level Diagram

Below, the final high-level diagram can be seen.

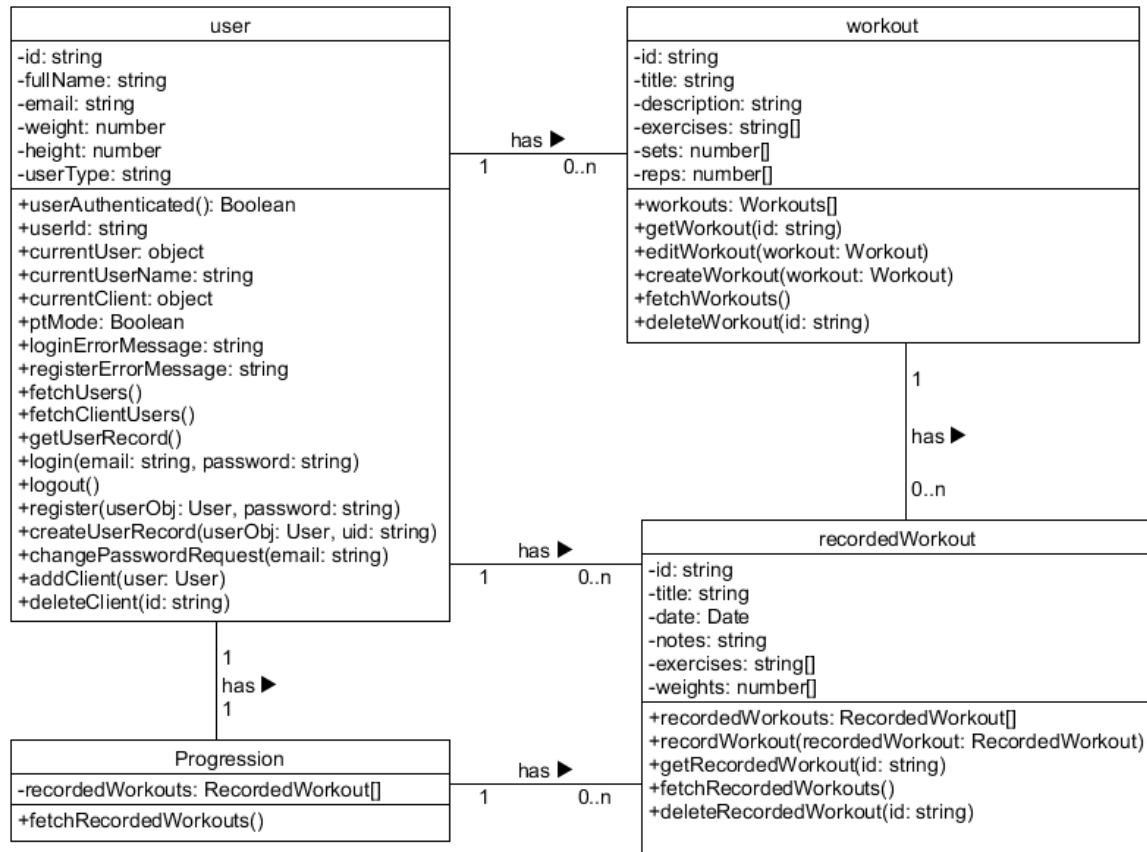The architecture contains various technologies:

- Android / IOS development, Xamarin.
- Flask, Python.
- Database, MySQL.
- Cloud Storage, Google Cloud Services / Amazon Web Services.
- Responsive Front-end, CSS3, Xamarin, Bootstrap.

2. High Level Diagram

3. Domain Model Diagram

4. Design Class Diagram

user

-id: string
-fullName: string
-email: string
-weight: number
-height: number
-userType: string

+userAuthenticated(): Boolean
+userId: string
+currentUser: object
+currentUserName: string
+currentClient: object
+ptMode: Boolean
+loginErrorMessage: string
+registerErrorMessage: string
+fetchUsers()
+fetchClientUsers()
+getUserRecord()
+login(email: string, password: string)
+logout()
+register(userObj: User, password: string)
+createUserRecord(userObj: User, uid: string)
+changePasswordRequest(email: string)
+addClient(user: User)
+deleteClient(id: string)

workout

-id: string
-title: string
-description: string
-exercises: string[]
-sets: number[]
-reps: number[]

+workouts: Workouts[]
+getWorkout(id: string)
+editWorkout(workout: Workout)
+createWorkout(workout: Workout)
+fetchWorkouts()
+deleteWorkout(id: string)

has ▶
1   0..n

1
has ▶
0..n

has ▶
1   0..n

recordedWorkout

-id: string
-title: string
-date: Date
-notes: string
-exercises: string[]
-weights: number[]

+recordedWorkouts: RecordedWorkout[]
+recordWorkout(recordedWorkout: RecordedWorkout)
+getRecordedWorkout(id: string)
+fetchRecordedWorkouts()
+deleteRecordedWorkout(id: string)

1
has ▶
1

Progression

-recordedWorkouts: RecordedWorkout[]

+fetchRecordedWorkouts()

has ▶
1   0..n

5. Project Structure

Using Ionic Angular for this project, the framework supports two main ways of sorting pieces of functionality within the project. These are Models and Services.

5.1. Models

Models are comparable to the object classes used in many objects orientated languages. Using a model allows me to create a reusable object which I can use to store and process vast amounts of information of the same category. An example of this within the project is a Workout model which is used to store all the data for a given workout.

5.2. Services

Services are used to create distributable and reusable functionality in a single place which then can be called upon by various page controllers. This again is very comparable to how an object orientated language works. This gives the developer the ability to create reusable functions that might be called on multiple pages allowing for better and cleaner programming practices. Within this project and example of this would be the use of the user service to handle all functionality such as login, forget password and so on, of the user.

6. Sequence Diagrams

6.1. UserService
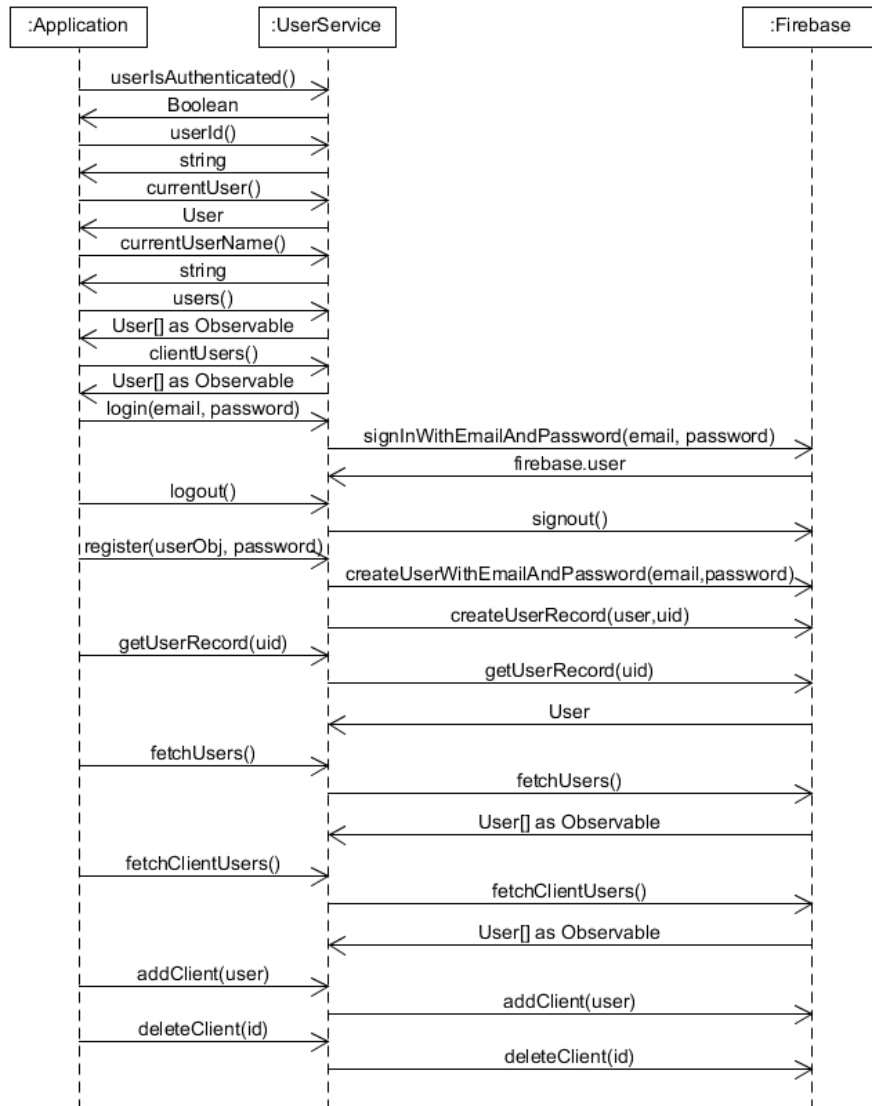
The User Service is used to connect and control functionality between the application and the backend in relation to the given logged in Users and anything User related. This is done through a mixture of module calls relating to the firebase authentication module, as well as the angular http client module. So not only does this service control all authentication measures for the application but it also controls all data retrieval and sending to do with users such as creating user records and clients.

6.2. WorkoutService
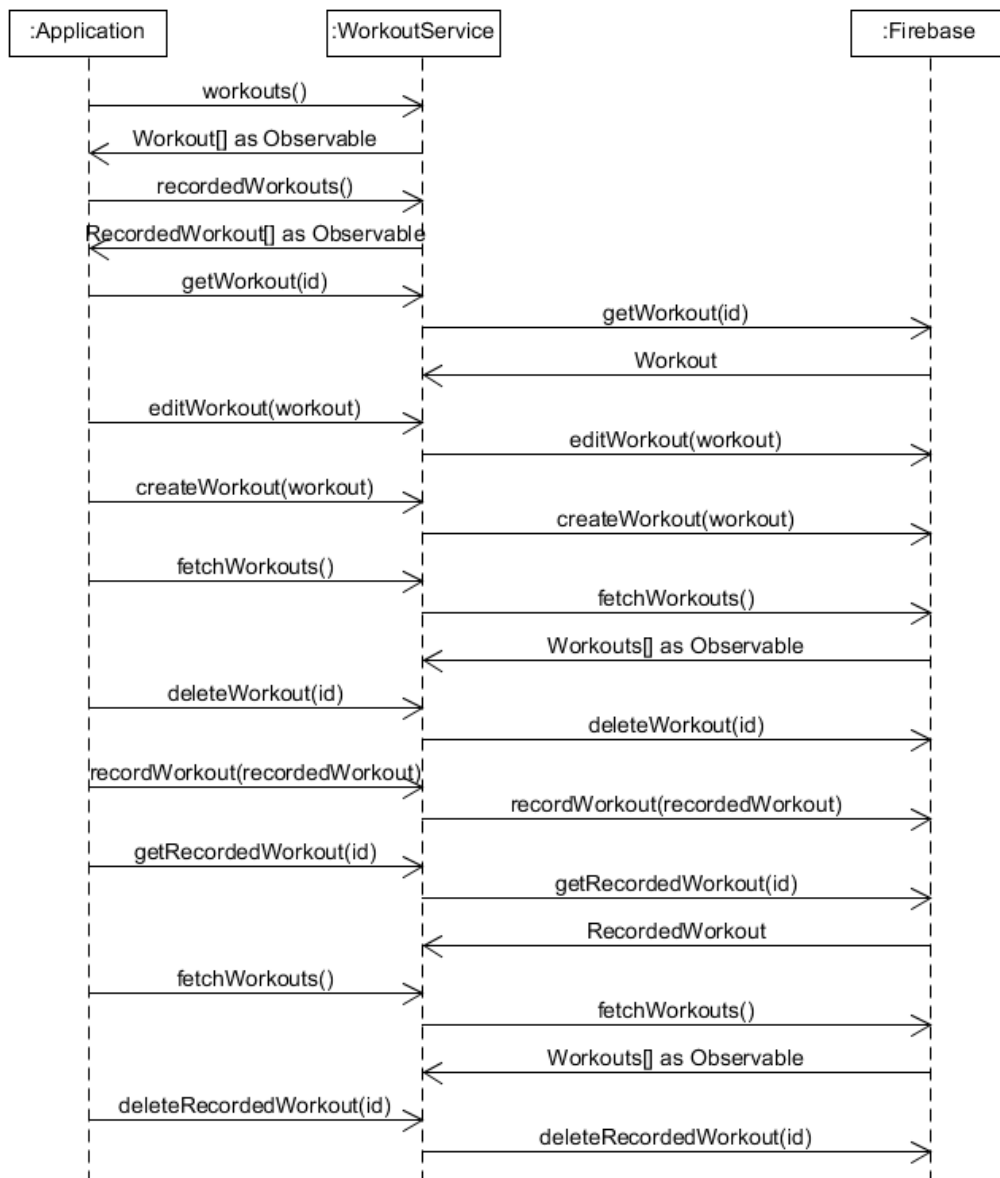
Like the User Service, the Workout Service handles all things related to workouts, connecting and communicating with the backend. In this service, the http client module from angular is the most used as it is the main form of communication used to the backend. So, when the application wants to create a new workout or record a workout it will all be handled through this service.

7. User Interface

The approach taken in developing the user interface for this application was to be simplistic and functional, providing a quick and easy to use experience for the users, clients and personal trainers to allow focus on their main goal of getting fitter.
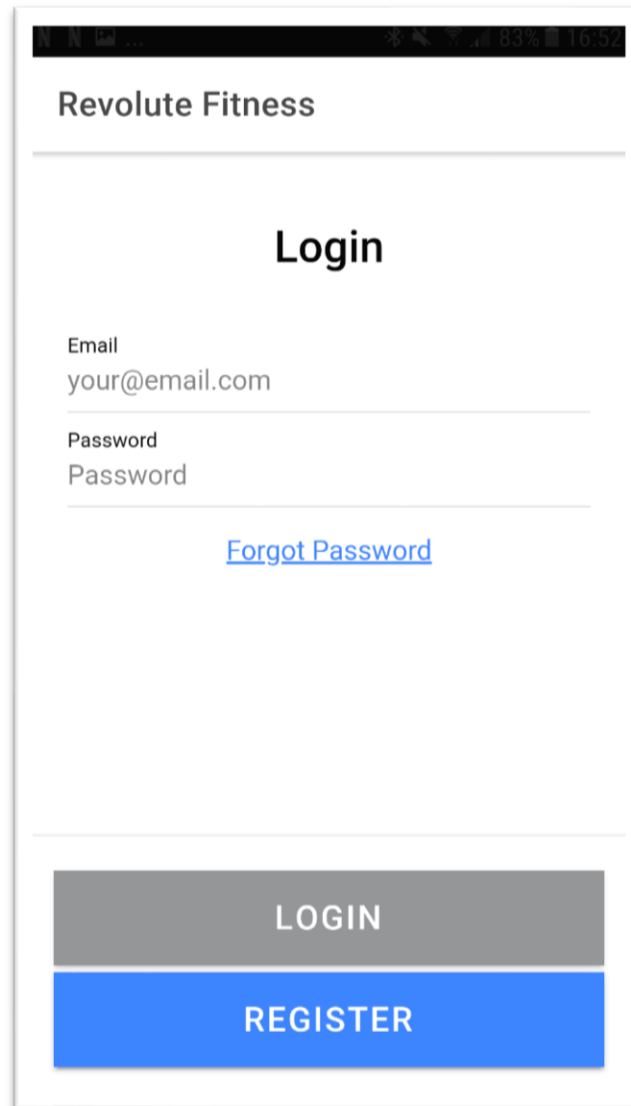
With using Ionic as the framework for this application, it meant being able to use web technologies such as HTML and CSS to create the user experience. Ionic with its premade components provides an already beautiful but simplistic and clean user experience and design.

One of the factors in the design of the UI was that it needed to be upheld through out the application, meaning that all data entry forms have a similar experience and all data display screens carried similar design as well. This allows the user to gain a quick understanding of how to use the application which applies throughout.

Another factor which was researched extensively was the main dashboard, instead of having the general slide out menu, it was opted to have one page with all the navigation of the app and provide adequate navigation through related pages. This is a very common practice with fitness apps today as it provides a very simple way to navigate giving the user more time to focus on their actual fitness.

7.1. Login

The login screen has a simplistic approach with easy but contrasting colours to aid with people with accessibility needs. Use is simple enter email and password and the login button enables allowing user to login.
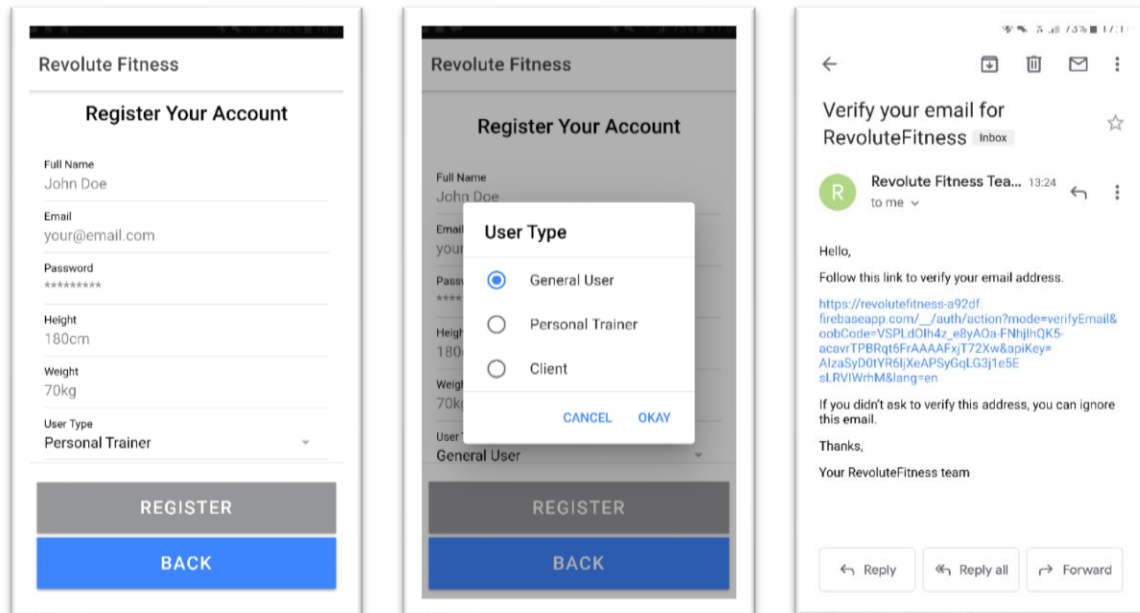
7.2. Register

The register screen is also very simplistic with simple well described input fields and even picklists where possible. When the user signs up supplying the needed data, they will receive a verification email before they can login.

7.3. Forgot Password

The forget password screen again follows the same style as pages before and is simplistic in design. The user submits their accounts email and they will receive and email to create a new password.
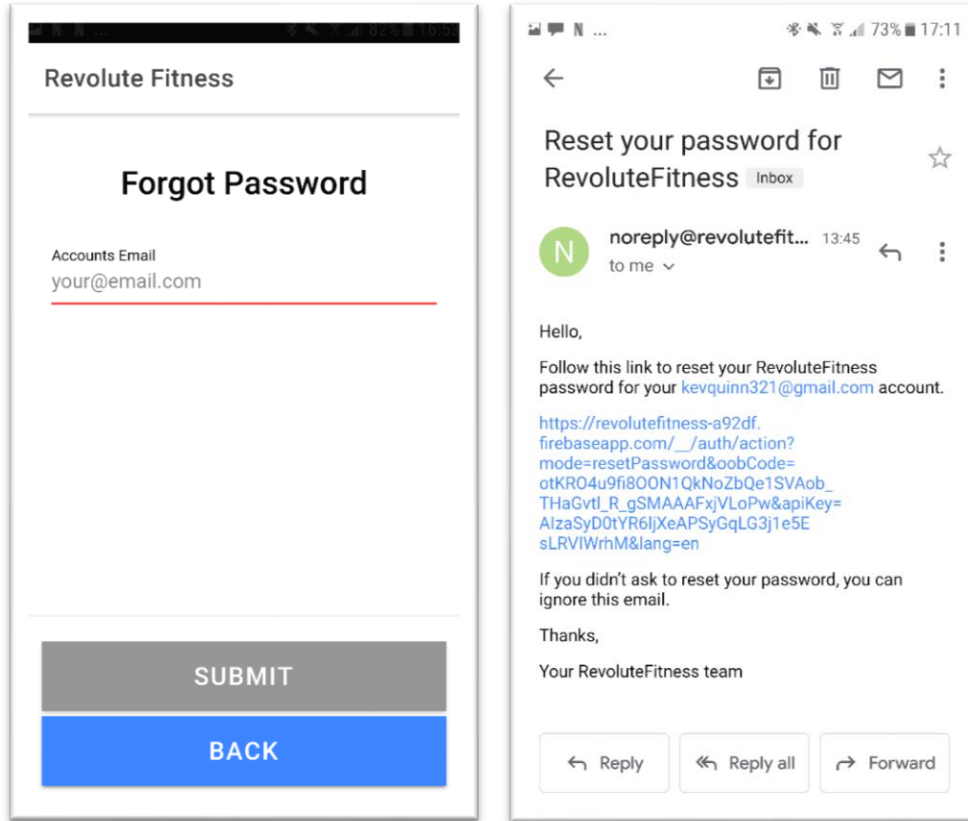
7.4. Main Dashboard

The main dashboard as discussed was designed to be the central focus for navigation around the application making it very simplistic allowing the user to better use their focus elsewhere.
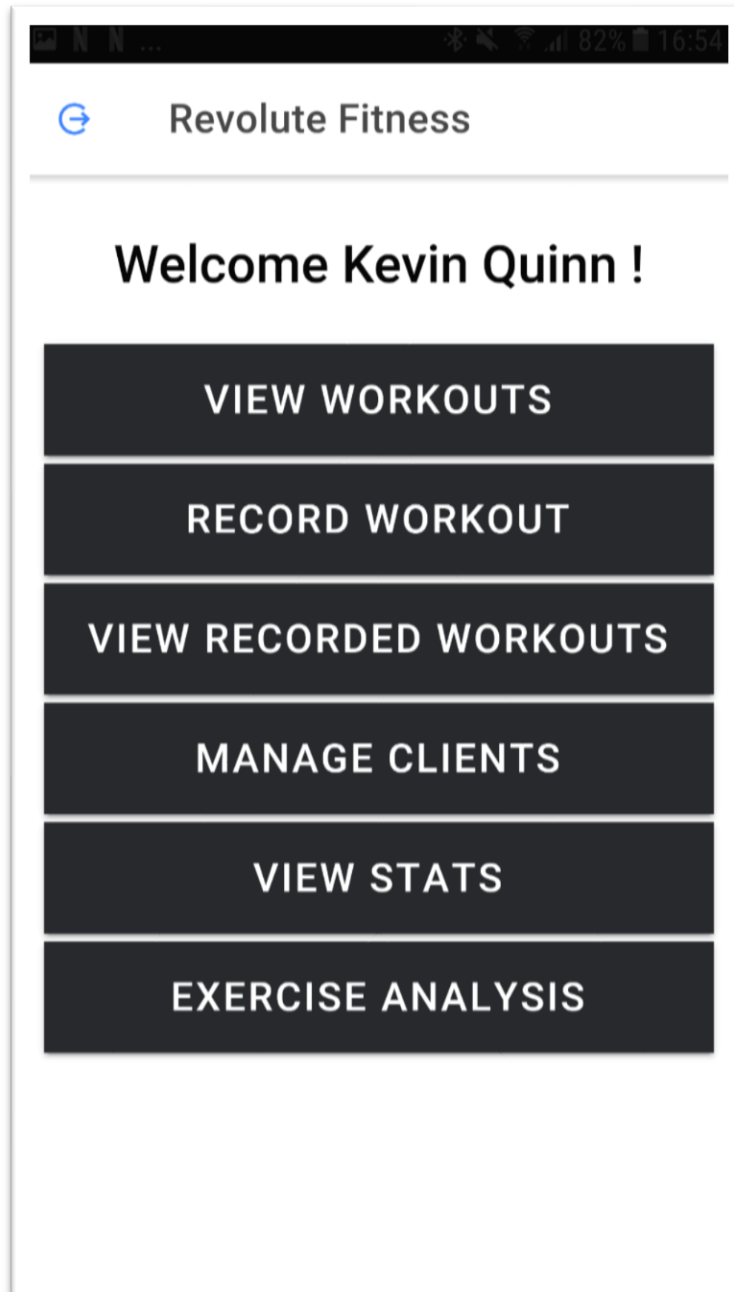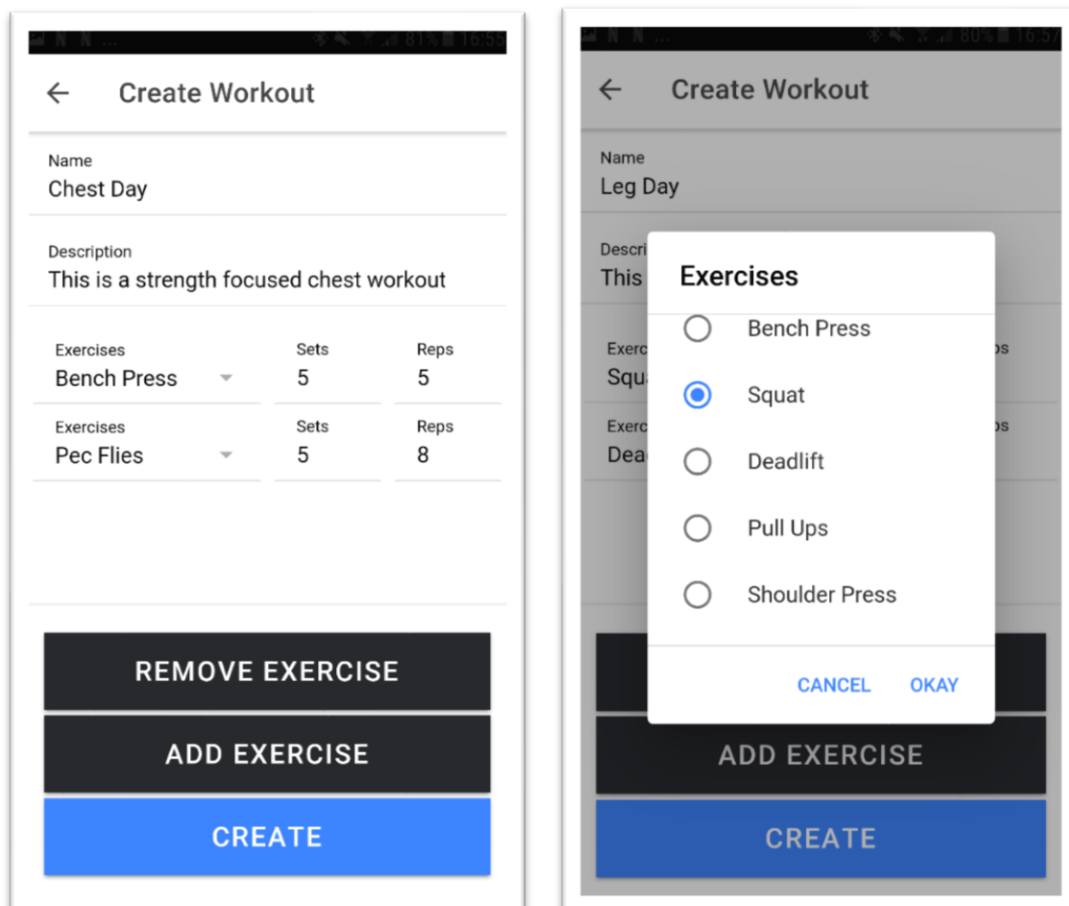
7.5. View Workouts

The view workouts screen is the users central place to manage their workouts, as can be seen a message is displayed to the user to "Add a Workout" when they currently have none which can be achieved through the plus in the top right corner. The user also can delete which can be done with a swiping option as seen.
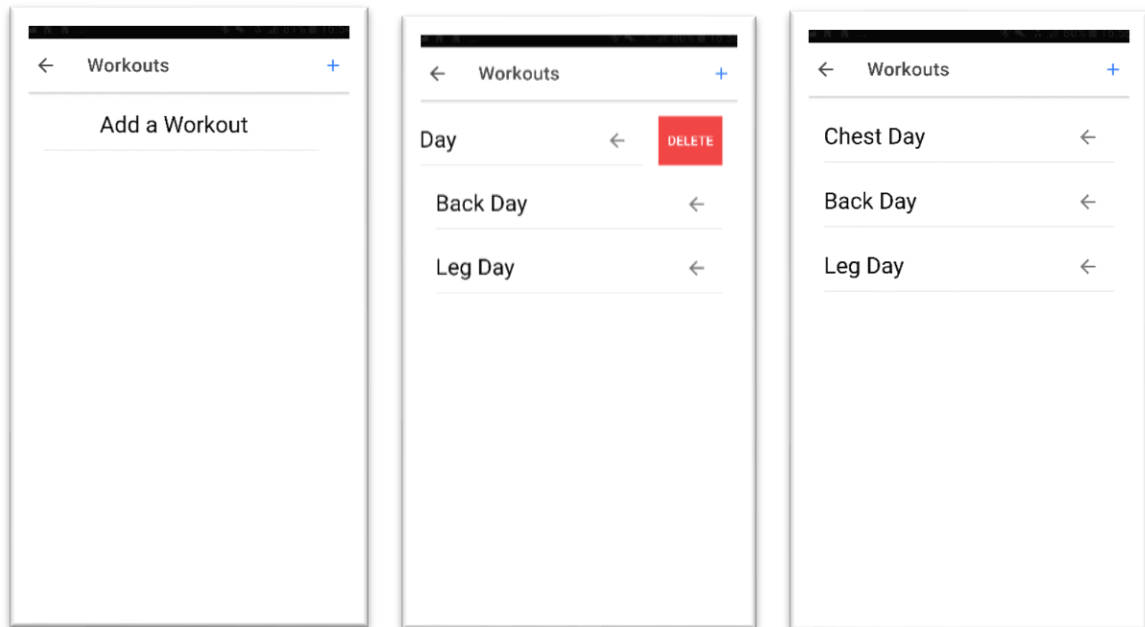
7.6. Create Workout

The create workout page, can be again seen to follow the style of the app providing the user with a form to enter a workout name, description and add exercises from and extensive picklist of exercises. The page is also designed in a way to prevent human error so exercises can only be added if the exercise before has been set and the form can only be submitted should all the requirements be met.

7.7. Select A Workout to Record

The select workout screen is similar in approach to the view workouts as it provides a list of the workouts available to record, following the simplistic style and design of the application.
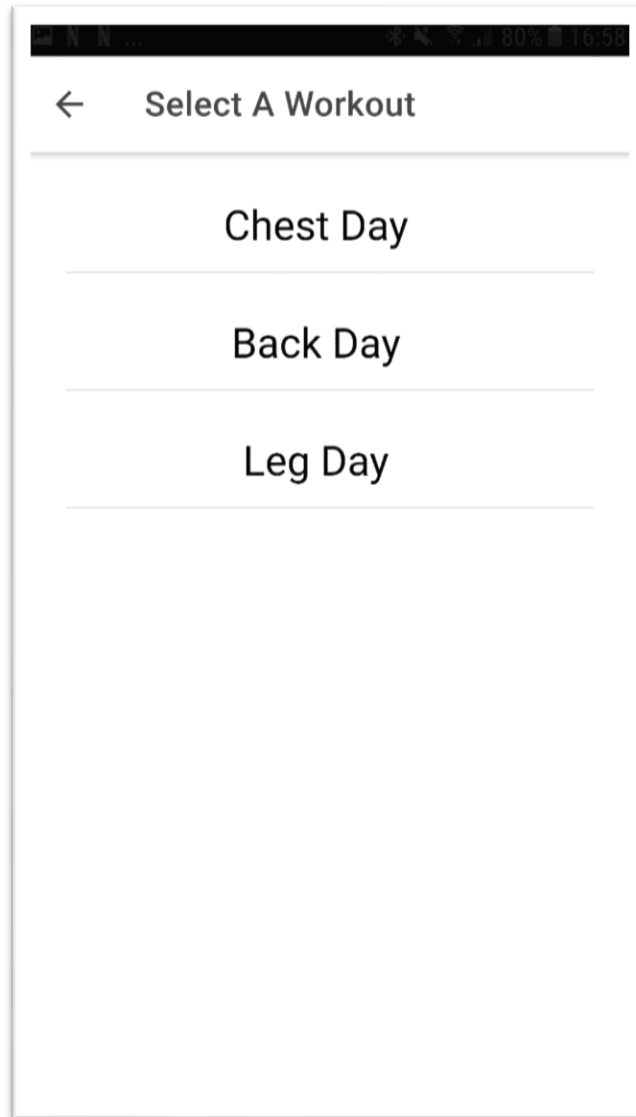
7.9. Record A Workout

The record a workout screen provides the user with the functionality of recording their workout giving them a notes field to keep track of important things that happened in a workout and also the ability to record the weight used for a pre set exercise in the workout again following the simplistic style of the application.

7.10. View Recorded Workouts

The view recorded workouts screen provides a list of workouts already been recorded by the user to give them the ability to look back at previous workouts. Displayed is the workout title and date of time of it being recorded to allow the user to be able to find the exact workout they want. Again, the sliding option is used to allow the user to delete a specific workout.
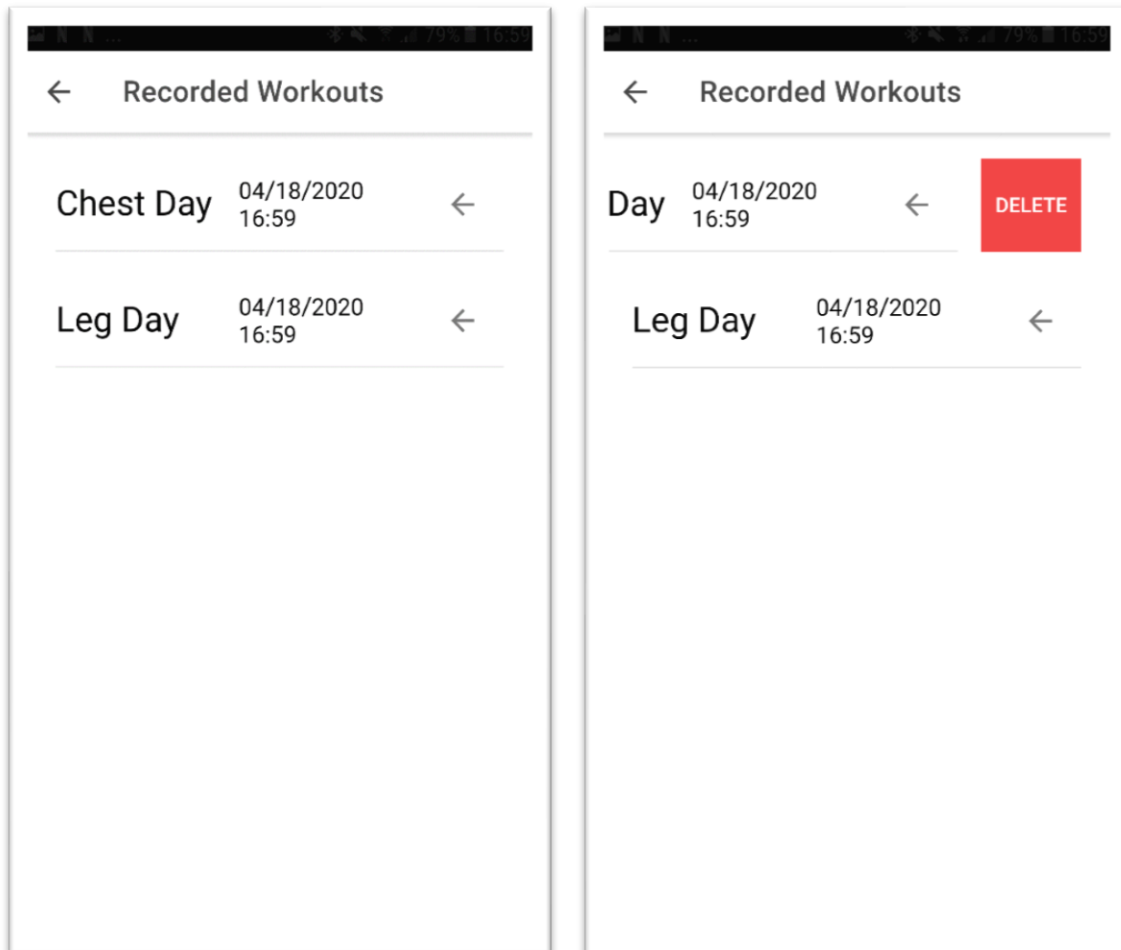
7.11. View Recorded Workout Details

The view recorded workout details screen is essentially the same as the record workout screen again following the same design and keeping it a simplistic approach.
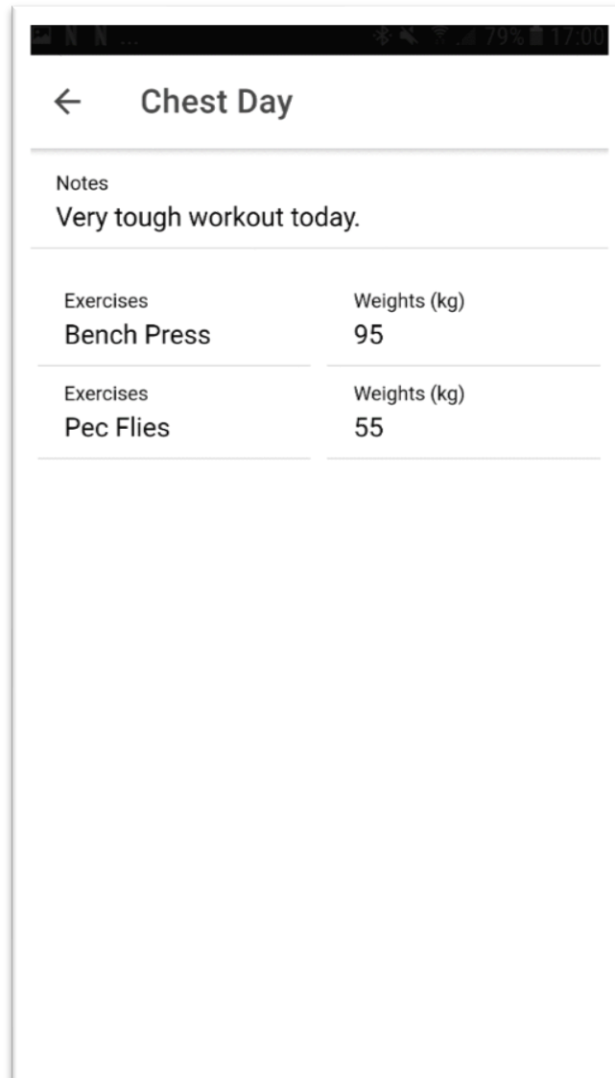
7.12. Manage Clients

The manage clients screen is only available to Personal Trainer users and gives them the ability to add a client by their email and then be able to enter the manage client's dashboard via clicking on the client or choosing to delete them by using the sliding option.

7.13. Manage Clients Dashboard

The manage client's dashboard is the view that allows a personal trainer to interact with their client. Giving them the ability to create workouts, view recorded workouts, and view their progression. It again follows the design approach of this application.
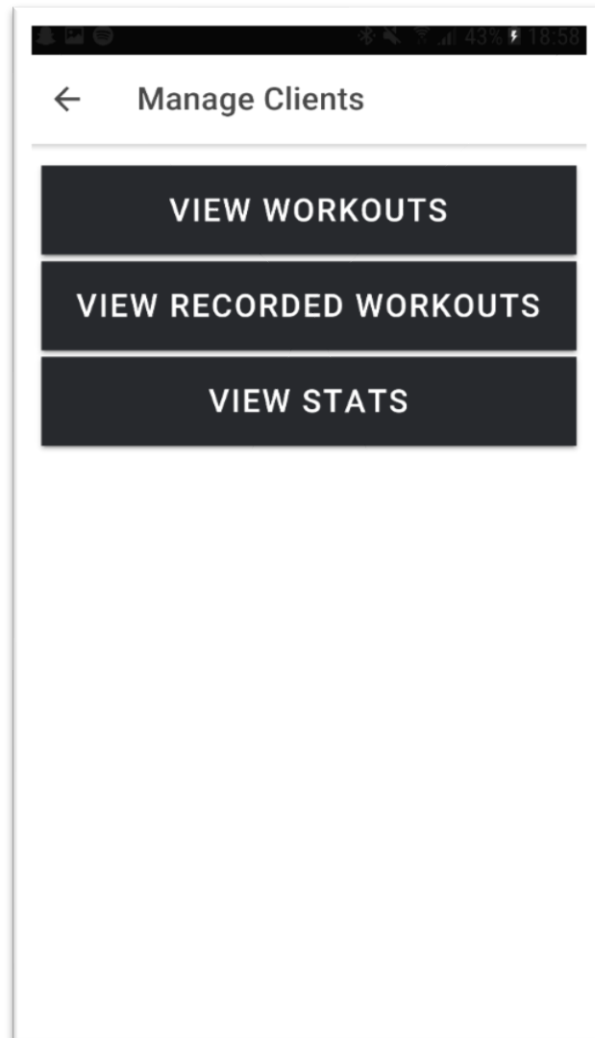
7.14. View Stats / Progression

The view stats / progression page provides the user with an interactive graph which will show them their overall progression using data from recorded workouts. It again is simplistic easy to understand and clean, again following the design principles of the application.
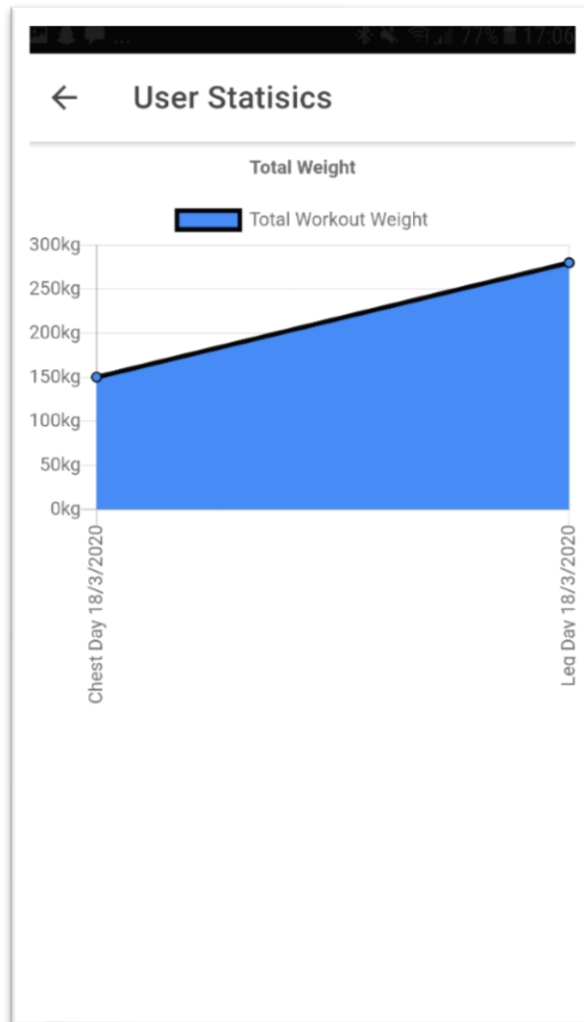
7.15. Exercise Analysis

The Exercise Analysis screen is a part of the research side of this project into the possible implementation of machine learning to aid in fitness. Thus, it is not as clean as it is not using ionic but still follows the approach of being simplistic to use.
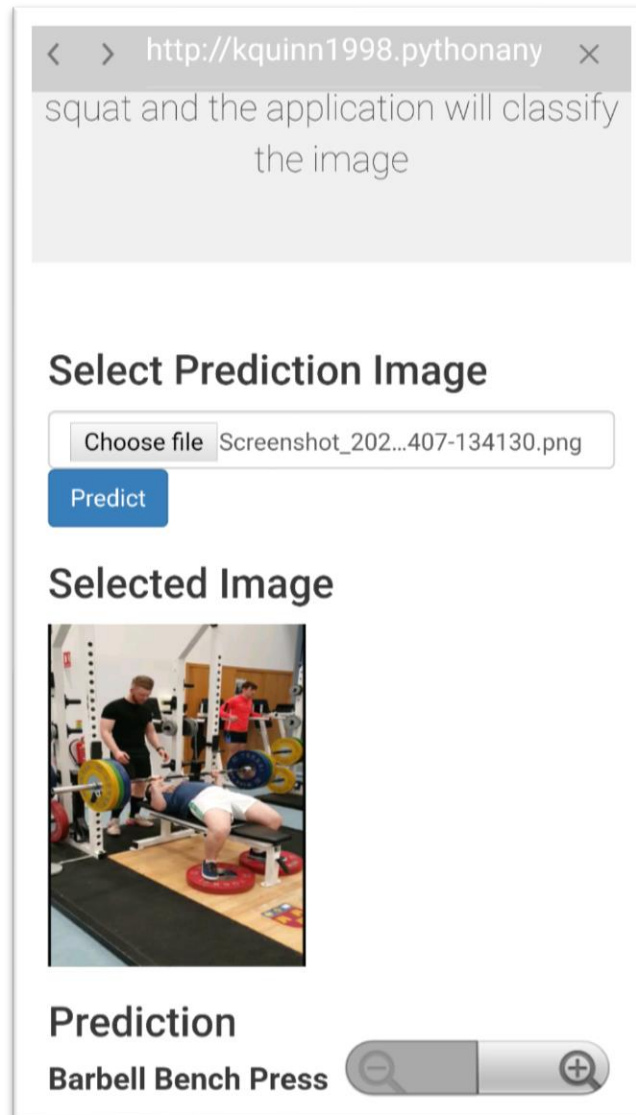
8. Appendix

8.1. Declaration

- I declare that all material in this submission e.g. thesis/essay/project/assignment is entirely my/our 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: Kevin Quinn

Student Number: C00216607

Signature:

Kevin Quinn

Date: 20/04/2020