Ollscoil Teicneolaíochta an Oirdheiscirt
South East Technological University

Final Year Project, Design Document, Gym Personal Training & Analytics App

Author – Daniel Murphy

Student No. – C00247818

Supervisor – Greg Doyle

Contents

Table Of Figures

1. Abstract

The purpose of this project is to create a cross platform (Android & iOS) application which will enable personal trainers or coaches to interact with their clients or players via mobile. They can manage their client's workout plans and see how they are progressing with each workout completed. They will also be able to provide meal plans and nutritional advice. Additionally, users will be able to create and edit their own plans using the app without interacting with a coach or personal trainer. Machine learning will also be implemented to perform different methods of statistical analysis and try predicting users' goals and give relevant feedback to improve time spent in the gym and the ultimate goal of using this app, to get healthier.

This document will describe the design of the application and show how it will be used. It will provide information on the relationships between models and the architecture of the system.

2. Introduction

This document was created after the research manual and functional specification so that the design requirements necessary to achieve the functional specification may be properly understood. The research manual discusses potential technologies as well as design ideas for User Interface (UI)/User Experience (UX) from existing/similar apps. This Design manual will use this information to set out exactly what technologies will be used to construct this project, as well as 'prototypes' of prospective screen designs, though these may differ once further development begins.

The first section explains the technological toolchain and how each chosen technology fits into this project. This follows the research on possible technologies and therefore addresses the advantages and suitability of utilising the technology.

The second section will show the architecture of my system and how the chosen technologies interact with each other. This will lead into the third section which will show how the Firestore database is structured and how the models are stored.

Furthermore, the last two sections will show the class diagram, sequence diagrams and the UI/UX prototype screenshots. Class diagrams represent the objects in our application and the different types of relationships with each other. They are also referred to as the blueprints of our system [1]. As for the prototype screenshots, these are just preliminary versions and are open to change. They are included to provide an idea of what the application will look like for the user.

3. Technologies

React Native and Firebase can be used together to create a comprehensive application. React Native allows for the development of native mobile applications using JavaScript and React, while Firebase provides several useful tools for building real-time and scalable applications. This project will also use Python to perform machine learning techniques and data analysis which will be used to give relevant feedback to the client. TensorFlow will be used to try and predict the user's goals.

In this project, React Native would be used to create the UI and handle the application's logic, while Firebase would be used to handle the application's data and user authentication. The Cloud Firestore is Cloud Firestore is a new, more intuitive data model that expands on the Realtime Database. Cloud Firestore also has more powerful, quicker searches and can scale further than Realtime Database [7]. It will be used to store workout and nutrition information, and the Firebase Authentication service will be used to handle user registration and login.

Firestore provides a way to store and retrieve user workout data, creating a seamless and efficient way for users to track their program and reach their fitness goals. Firestore will also be integrated to store user's images and videos which can be related to user's profile picture or an exercise. Firebase Cloud Functions can also be used to handle server-side logic and perform complex data processing tasks.

Finally, Python could be used to analyse the data collected by the app. For example, the data collected from Firebase can be exported to a CSV file, which can then be loaded into a Pandas DataFrame for data manipulation and analysis. Once the data is loaded into a Pandas DataFrame, various statistical techniques such as regression analysis and clustering can be used to analyse the data, extract insights, and show the user their progress.

3.1. React Native

React Native is an open-source frontend framework which was developed by Facebook. It is based on the JavaScript library React and allows for cross-platform mobile app development while housing all the code in a single codebase [2]. As the application will be cross platform, having the ability to write the code in a single codebase while the app being compatible for iOS and Android is a big advantage and will save a lot of time.

This project will also use Expo, which is an open-source framework built on and recommended by React Native. It is a development environment which allows for things like hot reload which shows changes to code instantly on devices and emulators. Expo will also manage any libraries and will work with Node.js to manage any Node Package Manager (NPM) libraries used in the project. This is where the UI/UX is created and will handle all the logic needed to navigate through the screens. This is also where the queries to the Firestore database come from.

3.2. Firebase

Firebase is a Backend as a Service (BaaS). This means it provides a connection to backend cloud storage and APIs. This project will also make use of the Firestore database. Firestore is a NoSQL document database that lets you easily store, sync, and query data which will be a vital part of this project. Firebase is built on Google's infrastructure and helps develop high-quality apps quickly by providing a variety of backend tools and services [3].

For this project, it will use different features Firebase offers. It will take advantage of the built in user authentication method. This allows the user to log in with an email and password. It also lets the user change their password or reset it via email link if they have forgotten it. Furthermore, when a personal trainer would like to sign up a client, an email is sent to the client with a link for them to sign up.

3.3. Machine Learning Technologies

Several machine learning technologies will be used in this project. Python, the programming language that Pandas and Tensorflow are built on, is a versatile programming language that is widely used for data analysis and machine learning. By using Python, we can easily integrate Pandas and Tensorflow into our application and perform various data analysis and prediction tasks.

Pandas DataFrame is a powerful data manipulation library for Python that allows for easy handling of large data sets. By using Pandas, we can easily extract, clean, and pre-process the data collected from the user to train and test machine learning models [5].

Tensorflow, on the other hand, is an open-source machine learning library that can be used to build and train neural networks. By utilizing Tensorflow, we can create models that can predict when the user is expected to hit their goal weight based on the data collected. Tensorflow also allows for easy deployment of the trained models on different platforms, including mobile devices and web services [6]. Scikit-Learn could also be used to train a linear regression model to predict these goals.

4. System Architecture

4.1. High-Level Diagram

The following high-level diagram shows the architecture of the system. The users are registered and authenticated with Firebase and React Native provides a UI so the user can easily manage their workouts, nutrition, clients or track their progress.

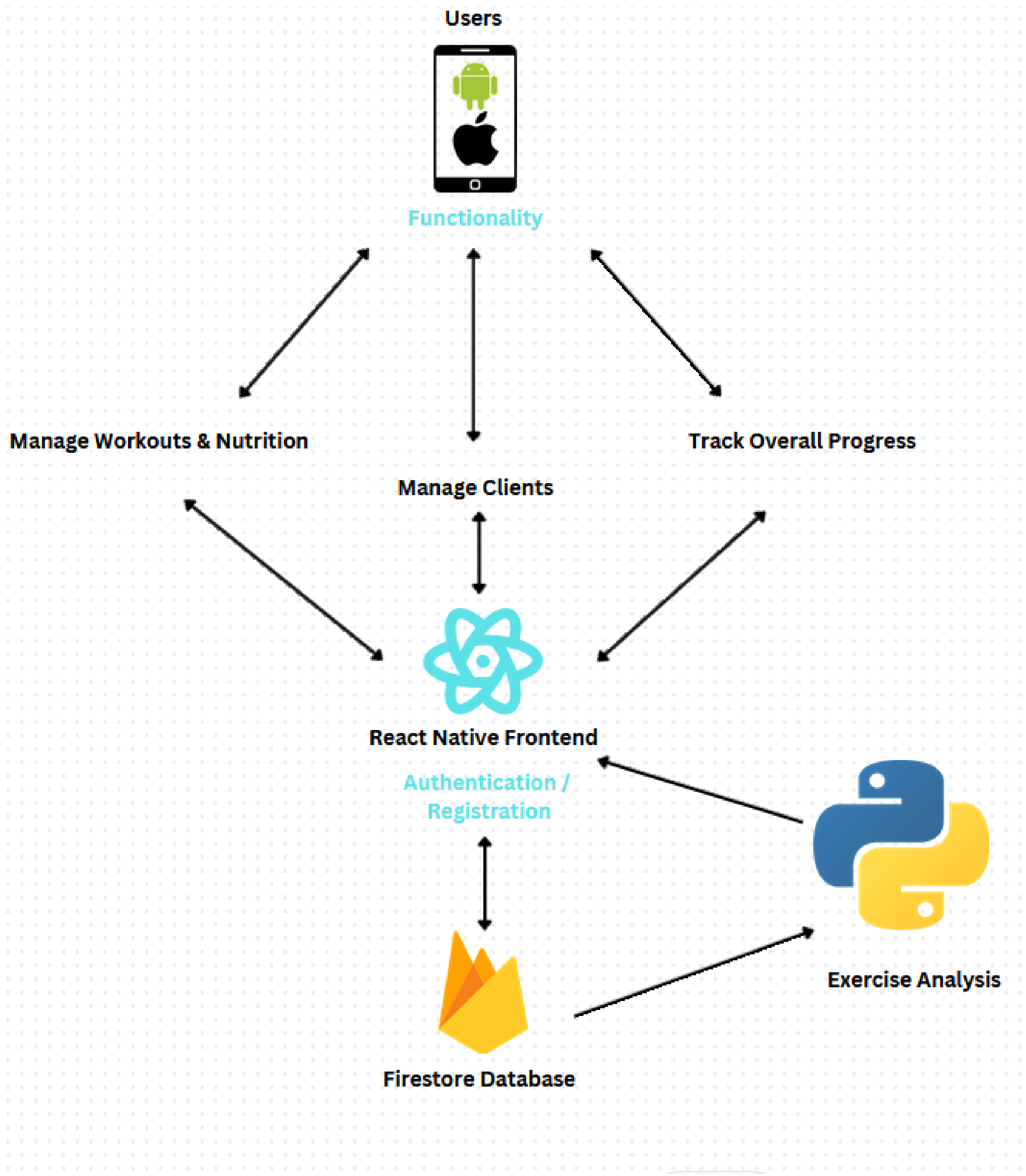


Figure 1 High-Level Diagram

4.2. MVC Architecture

MVC architecture (Model-View-Controller) is a popular architectural pattern that splits an application into three related components: the Model, the View, and the Controller [8]. The Model represents the data and business logic of the application, the View provides the user interface, and the Controller manages communication between the Model and the View. The separation of responsibilities provided by the MVC design makes the code more flexible and easier to maintain.

MVC architecture will be used in this application to manage data regarding the users' weight, workout routines, and nutrition plans. The View component would oversee the user interface, which would include displaying the weight graph, workout routine instructions, and nutrition recommendations. The Controller component would be in charge of controlling the Model-View communication, such as updating the weight data when the user records their weight and presenting the new weight graph in the View. The MVC cycle would ensure that the user's weight data is shown and updated properly, and that the user interface is responsive and intuitive.

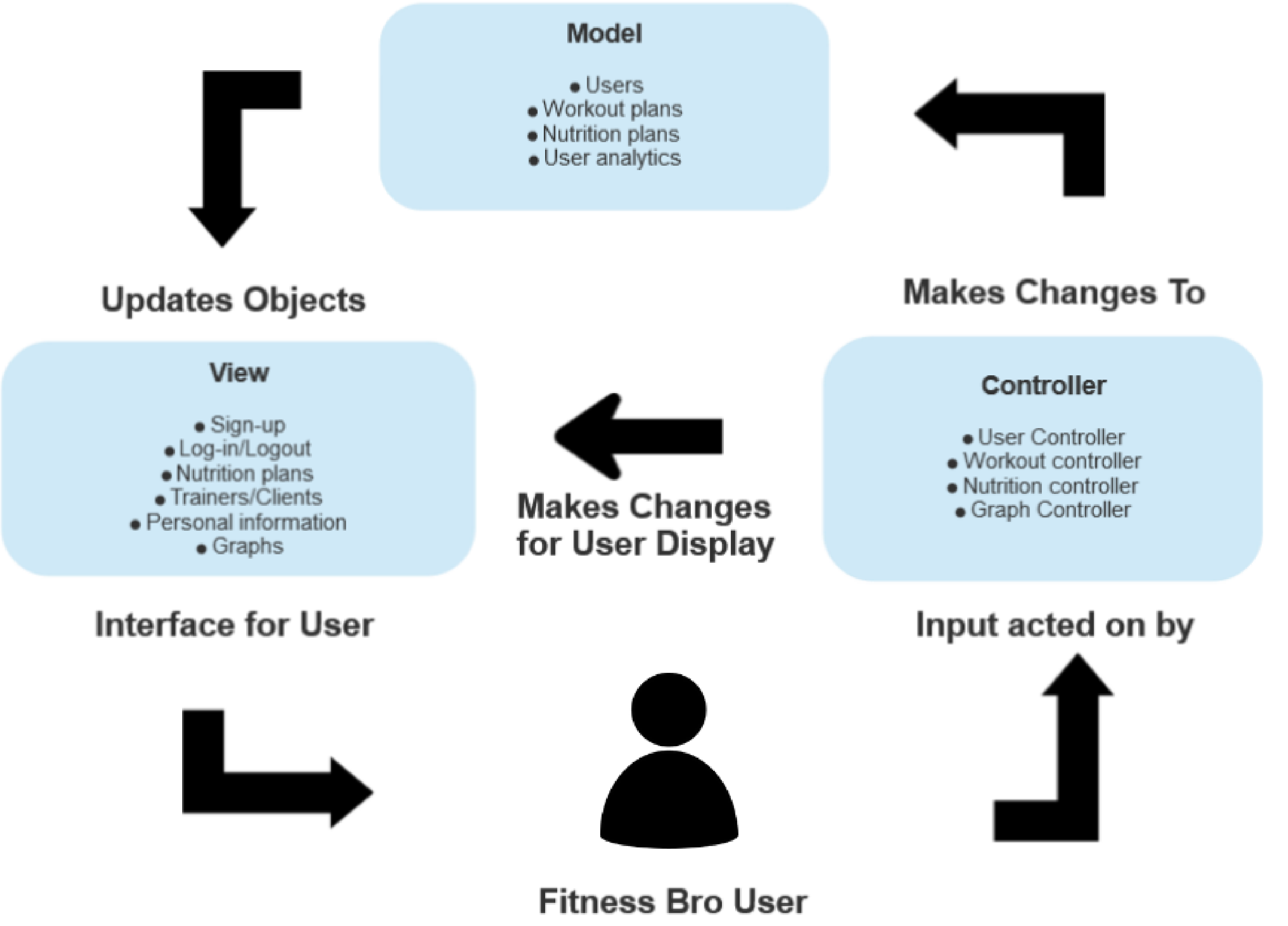


Figure 2 MVC Architecture

4.3. Screen Flow

The following figure represents the hierarchical layout which shows which screens can be navigated to and from. This layout is subject to change as development continues.

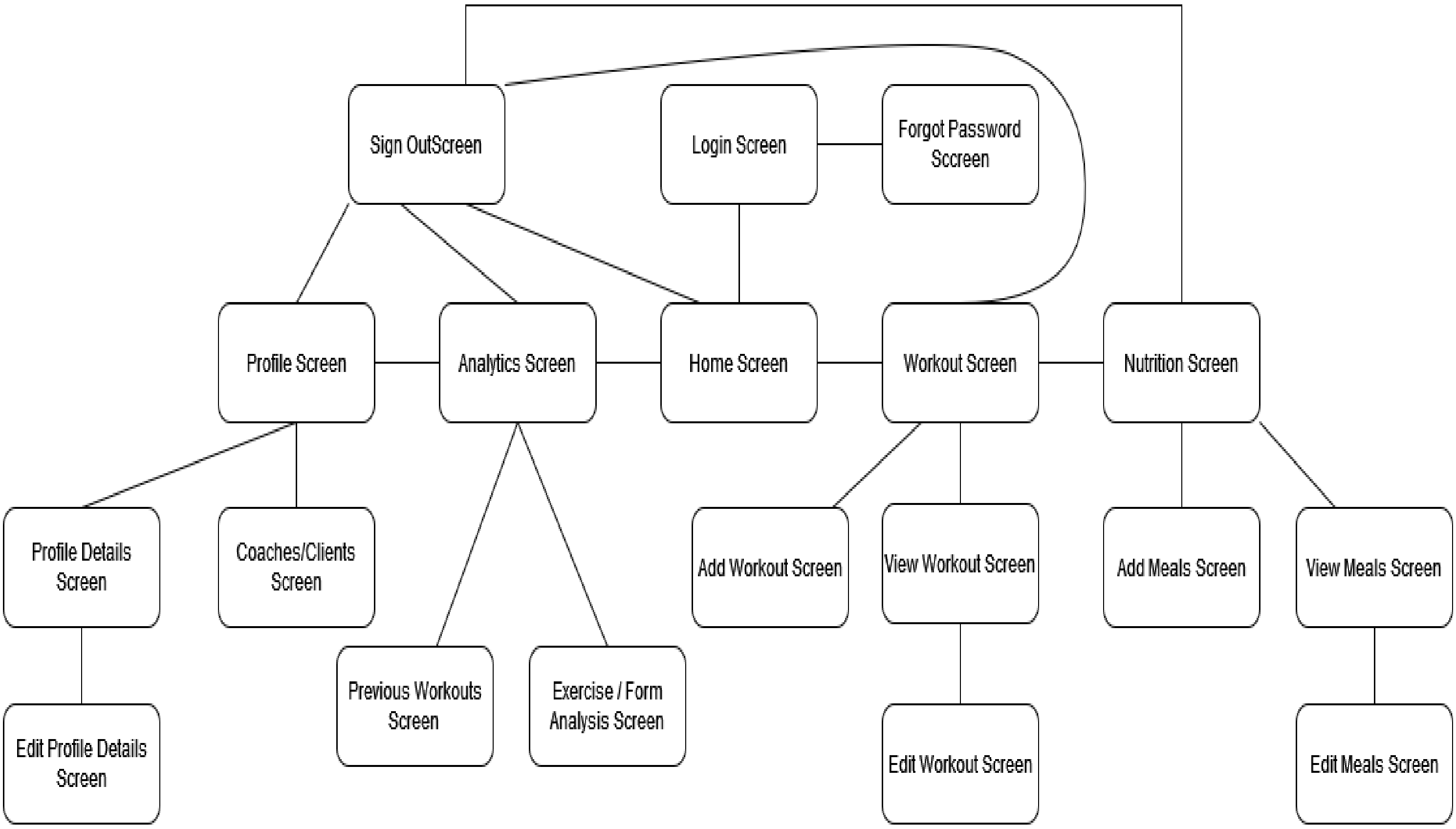


Figure 3 Screen Flow

5. Database

A Firestore database will be used to store and retrieve user data. The database consists of several collections, including a "users" collection to store user profiles, a "workouts" collection to store workout data, and a "nutrition" collection to store nutrition data.

Each user's data is stored within their own document within the "users" collection, with subcollections for workouts and nutrition. The "workouts" and “nutrition” collection will contain the documents belonging to the user after creating a workout or nutrition plan.

This structure allows for easy and efficient retrieval and manipulation of user data, as well as the ability to track and analyse trends over time. It allows the user to see only data they have created. By specifying firebase rules, we can restrict the user from seeing all the data in the database and show only data relating to their User ID. We can also query our assigned workouts / nutrition database by getting the workouts / plans relating to the client field. This will be the assigned user's email which will help in a query against the collection to return any workouts / plans with the user's email. This structure is demonstrated in the following database design:

Users [

<$UID>

- Email
- Name
- Current_weight
- Goal_weight
- Age
- Weights[
  - Date
  - Weight

  ]
- Workouts [
  - Id
  - Day
  - Name
  - Training_type
  - Exercise [
    - Name
    - Sets
    - Reps
    - Weight
    - VideoLink

    ]

  ]
- Nutrition [
  - Id

- Name
- Day
- Snack
- Meal [
  - Name
  - Calories
  - Carbohydrates
  - Fat
  - Protein

  ]

]

]

Teams [

- Id
- Name
- Members{
  - Name
  - Email

  }

]

Clients [

- Name
- Email

]

AssignedWorkouts [

- Id
- Client
- Trainer
- Name
- TrainingType
- Exercise [
  - Name

    - Sets
    - Reps
    - Weight
    - VideoLink
      ]

]

AssignedNutrition [

- Id
- Client
- Trainer
- Name
- Day
- Meal [
    - Name
    - Calories
    - Carbohydrates
    - Fat
    - Protein
      ]

]

6. Class Diagram

- Client/Trainer: This class will represent the user of the application, and will include attributes such as name, age, and weight as well as behaviours such as tracking workouts and nutrition and setting goals.

- Workout: This class will represent a user's workout, and will include attributes such as name, type, description, and duration as well as behaviours such as adding exercises and tracking progress and CRUD operations.

- Exercise: This class will represent the exercises performed during a workout, and will include attributes such as name, type, sets, reps, and weight, as well as behaviours such as tracking progress and CRUD operations.

- Nutrition: This class will represent the nutrition tracked by the user, and will include attributes such as name, description, and meals, as well as behaviours such as tracking intake and CRUD operations.

- Meals: This class will represent the meals a user eats in their day and will include attributes like name, calories, protein, carbs, and fat, as well as behaviours such as CRUD operations.

- Team: This class will represent a collection of users which is assigned to one trainer/coach. It will include attributes like name, workout, nutrition, and members, as well as behaviours such as viewing workouts/nutrition plans and managing clients.

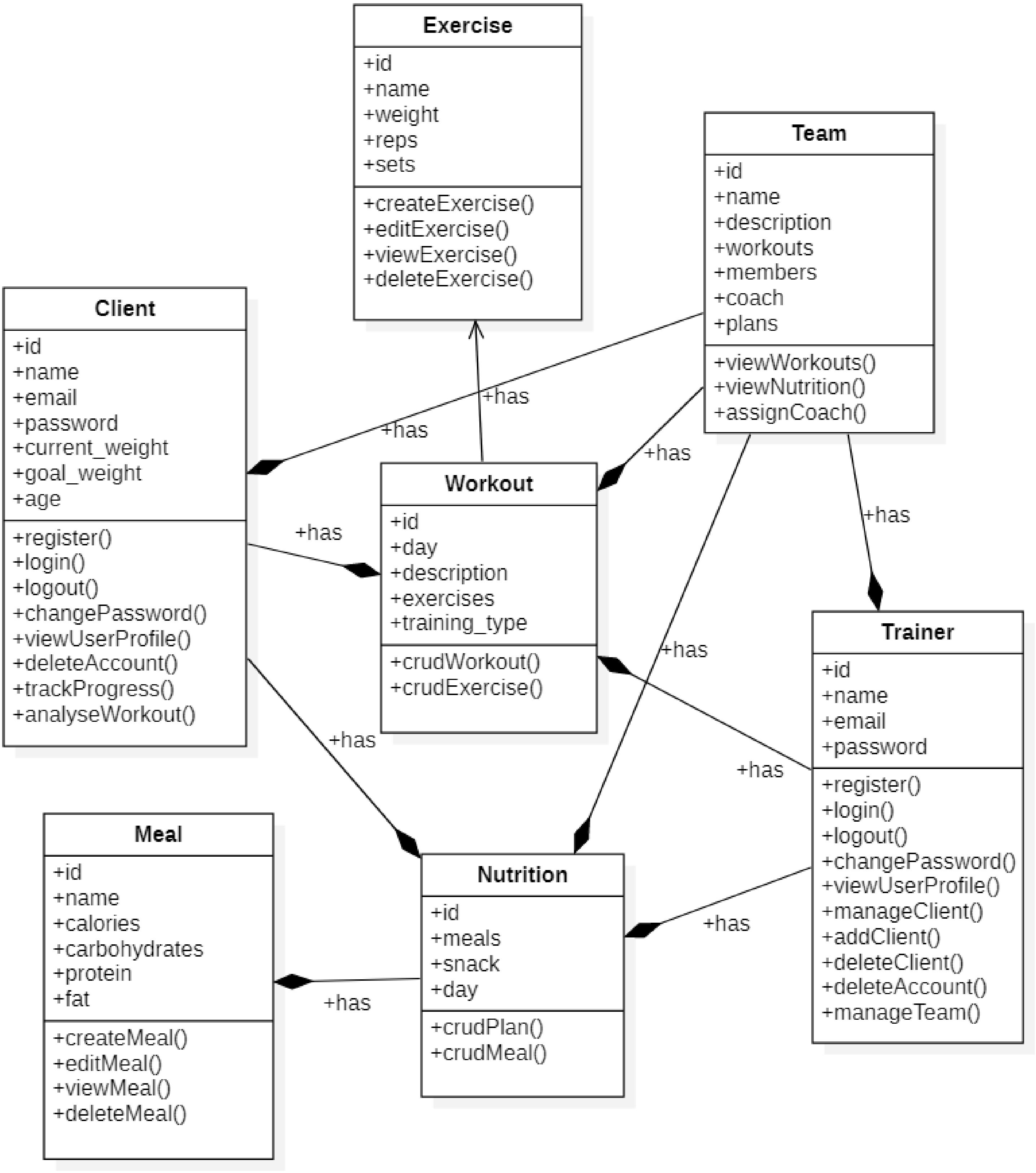


Figure 4 Class Diagram

7. System Sequence Diagram

The system sequence diagram shows the many interactions that take place between the user and the app to carry out activities such as logging in, registering, managing workout and nutrition data and managing clients. These interactions contribute to the user's ability to navigate and use the app to achieve their fitness objectives. The app will have key features such as user profile management, progress monitoring, and analytics in addition to the specific functionality depicted in the figure. These features, when combined, offer customers with a complete and user-friendly tool for managing their fitness journey. The system sequence diagram serves as a blueprint for the development team, ensuring that the app is designed to meet user needs in a streamlined and efficient manner.

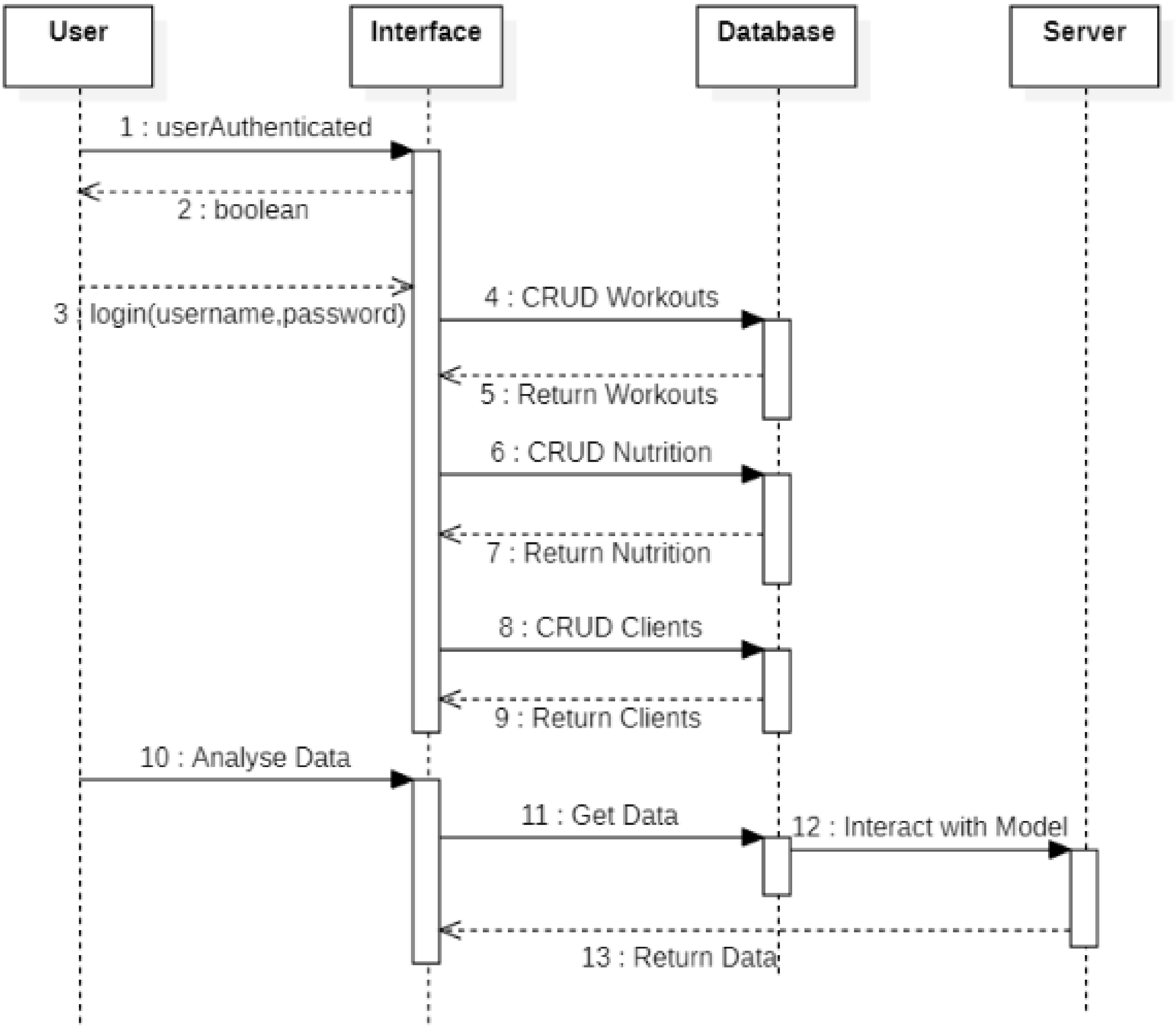


Figure 5 System Sequence Diagram

8. UI/UX Prototypes

8.1. Colour Palette

The colour scheme of the application features a combination of two main colours: #0792F9 and #4682B4 which are shown below. The first colour, #0792F9, is a bold and vibrant shade of blue, which gives the app a modern and energetic feel. When used as the primary colour of this app, it sets a positive and professional tone.

The second colour, #4682B4, is a more subdued shade of blue that provides a nice contrast to the boldness of the first colour. When used in combination with #0792F9, #4682B4 creates a professional and friendly looking UI. This colour will be used as an accent colour or as the background colour.

Overall, the colour scheme of this application is clean, modern, and professional. The use of different shades of blue creates a sense of depth and balance that makes the app feel cohesive and easy on the eyes.

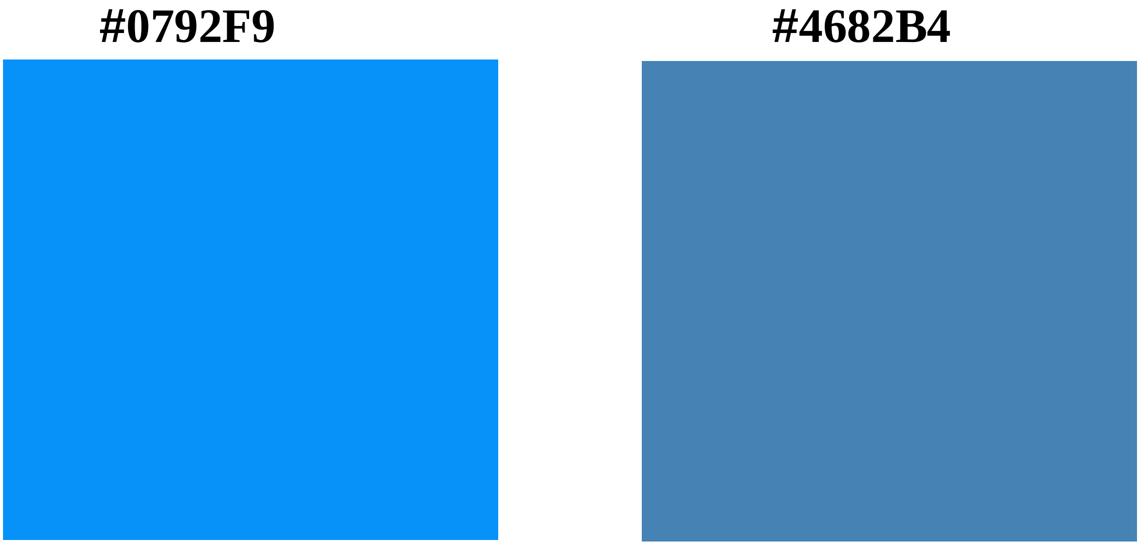


Figure 6 Colour Palette colours

8.2. Screen Prototypes

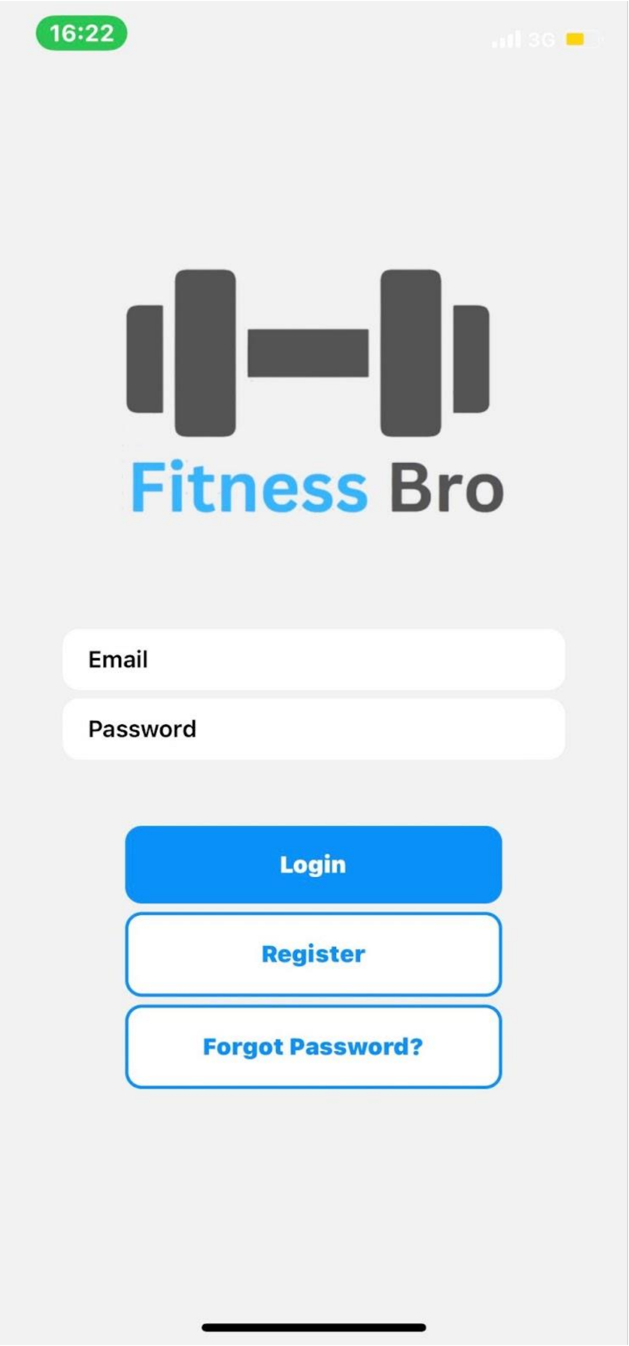


Figure 7 Prototype Login Screen

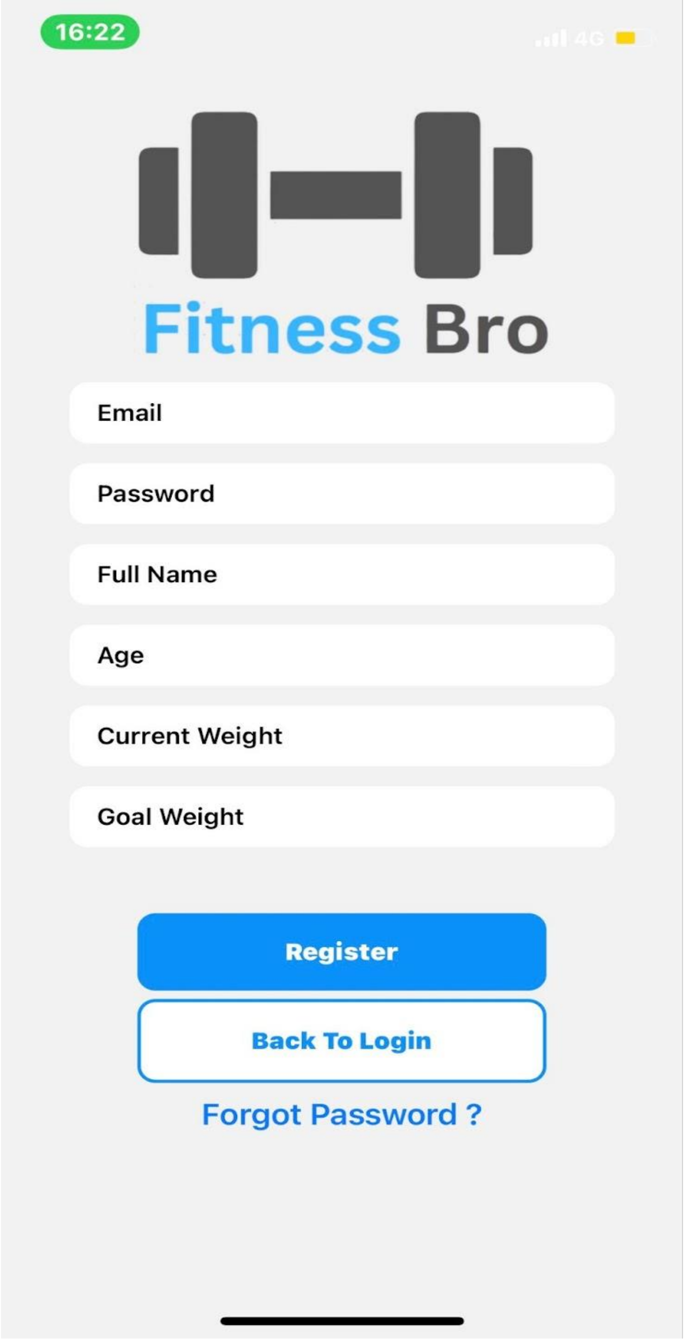
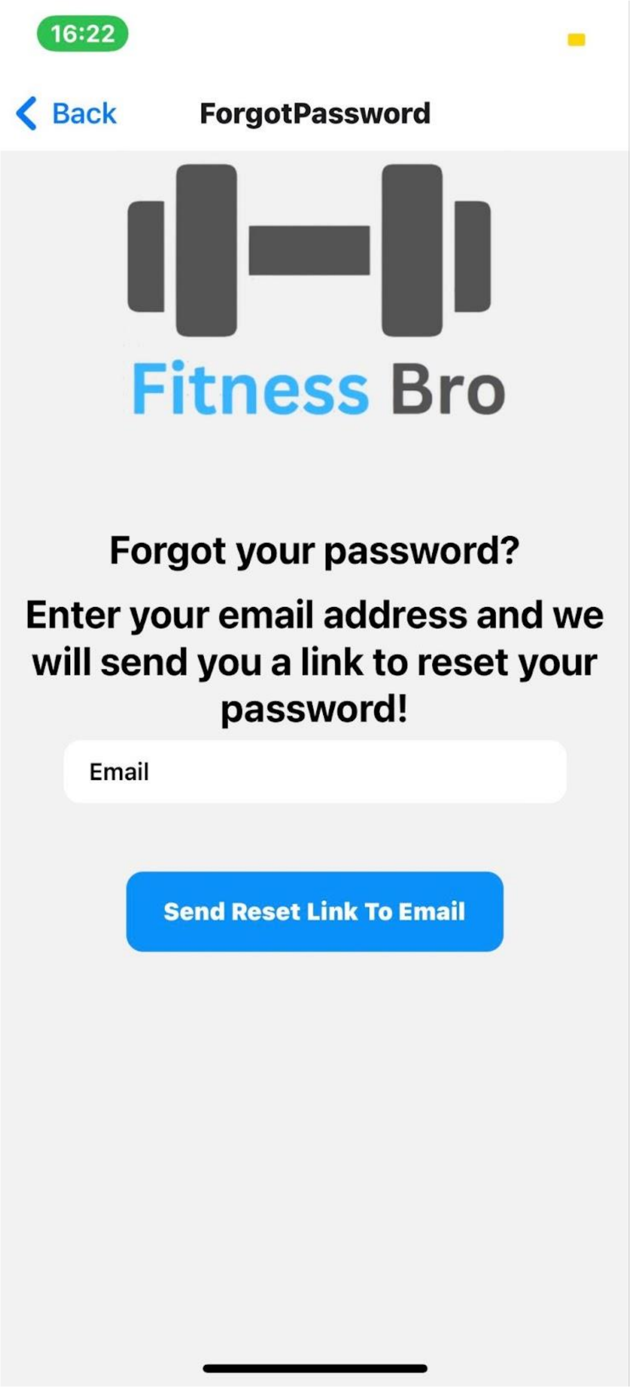


Figure 8 Prototype Register Screen



16:22

Back

ForgotPassword

Fitness Bro

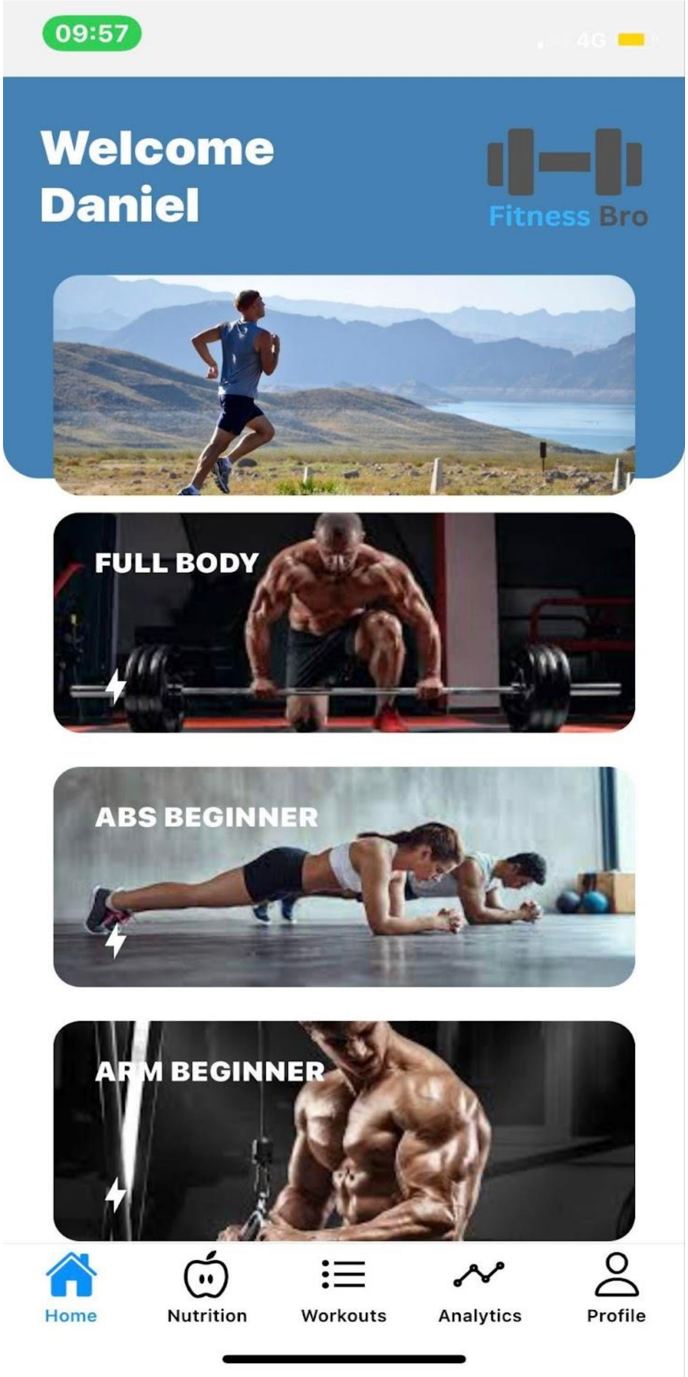
Forgot your password?

Enter your email address and we will send you a link to reset your password!

Email

Send Reset Link To Email

Figure 9 Prototype Reset Password Screen

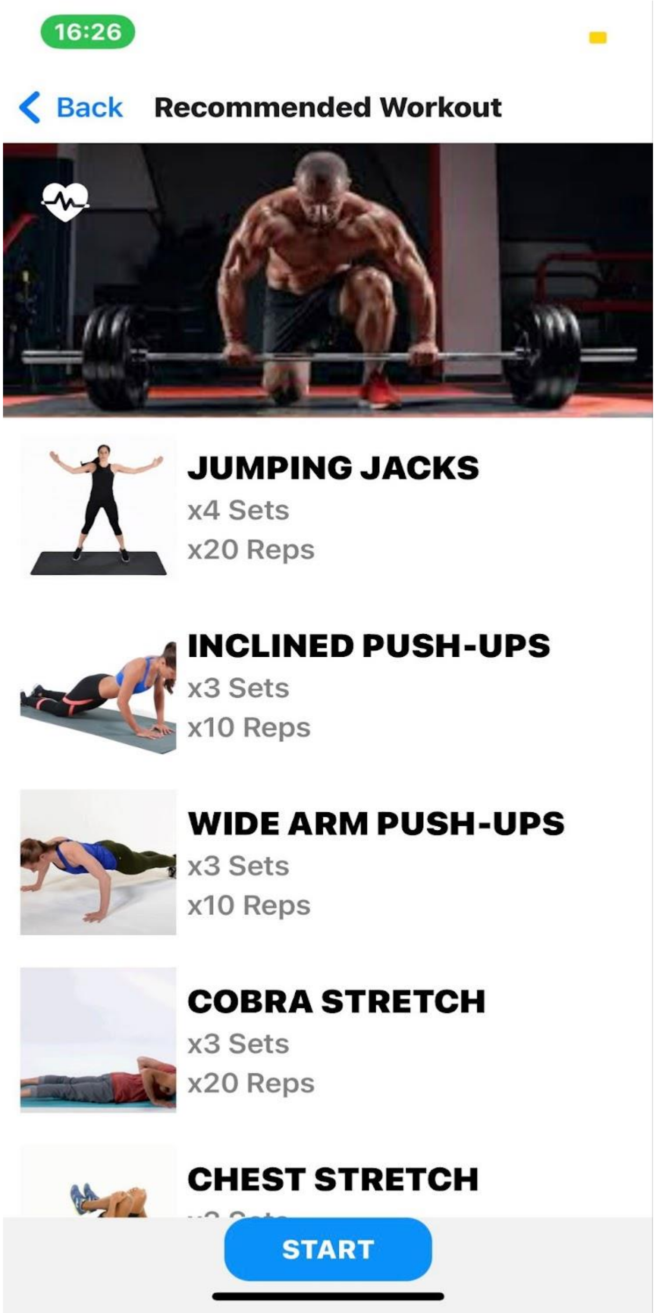


09:57

Welcome Daniel

Fitness Bro

Figure 10 Prototype Home Screen



JUMPING JACKS
x4 Sets
x20 Reps

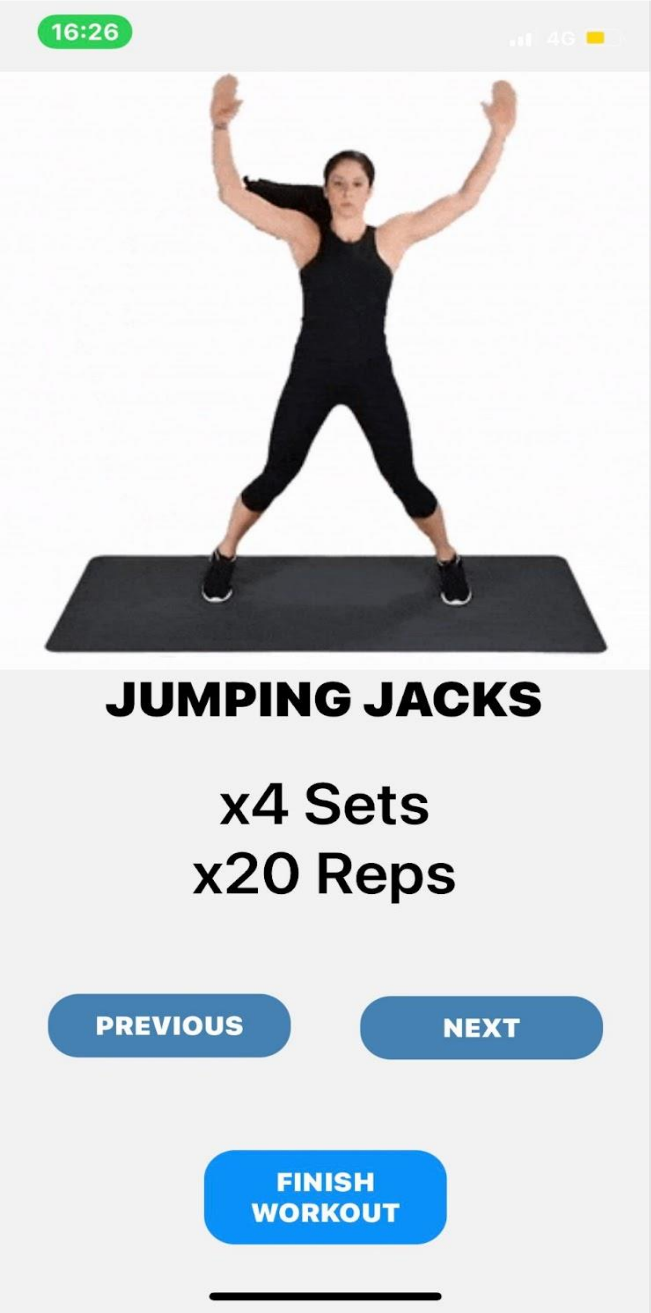
INCLINED PUSH-UPS
x3 Sets
x10 Reps

WIDE ARM PUSH-UPS
x3 Sets
x10 Reps

COBRA STRETCH
x3 Sets
x20 Reps

CHEST STRETCH

Figure 11 Prototype Workout Screen



JUMPING JACKS

x4 Sets
x20 Reps

Figure 12 Prototype Exercise Screen

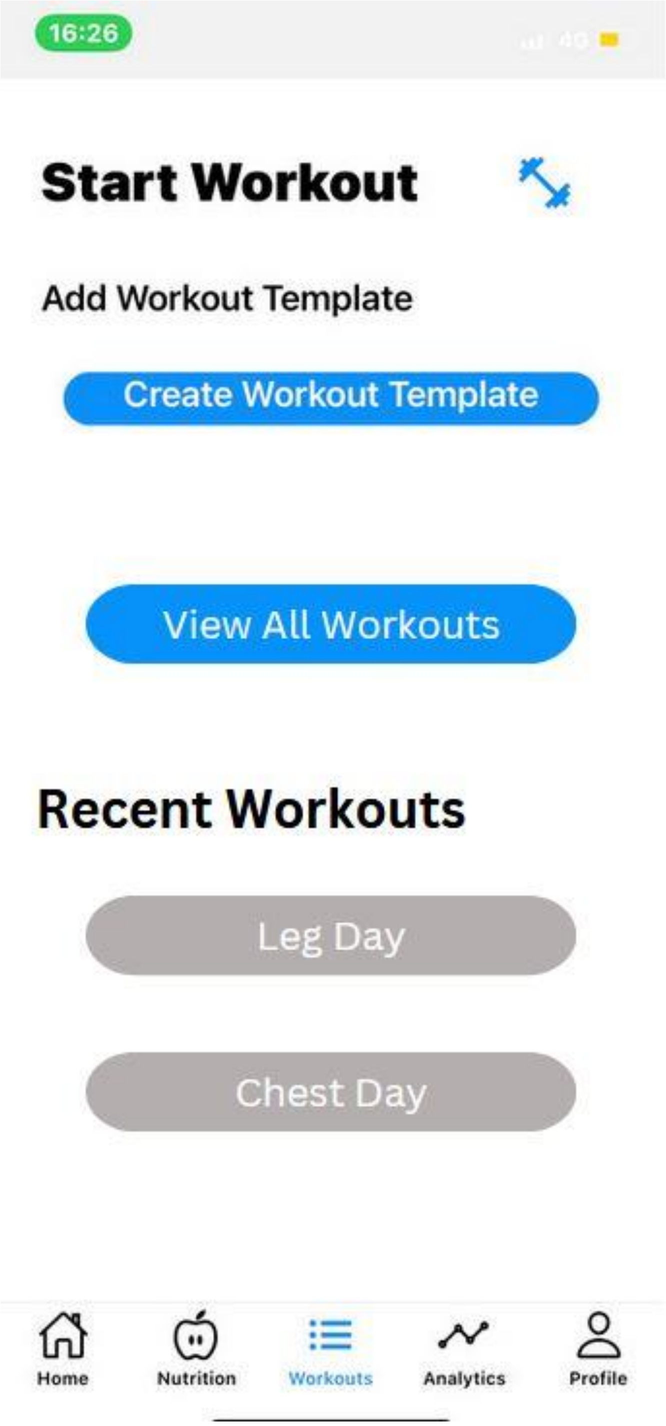


Figure 14 Prototype Workout Home Screen

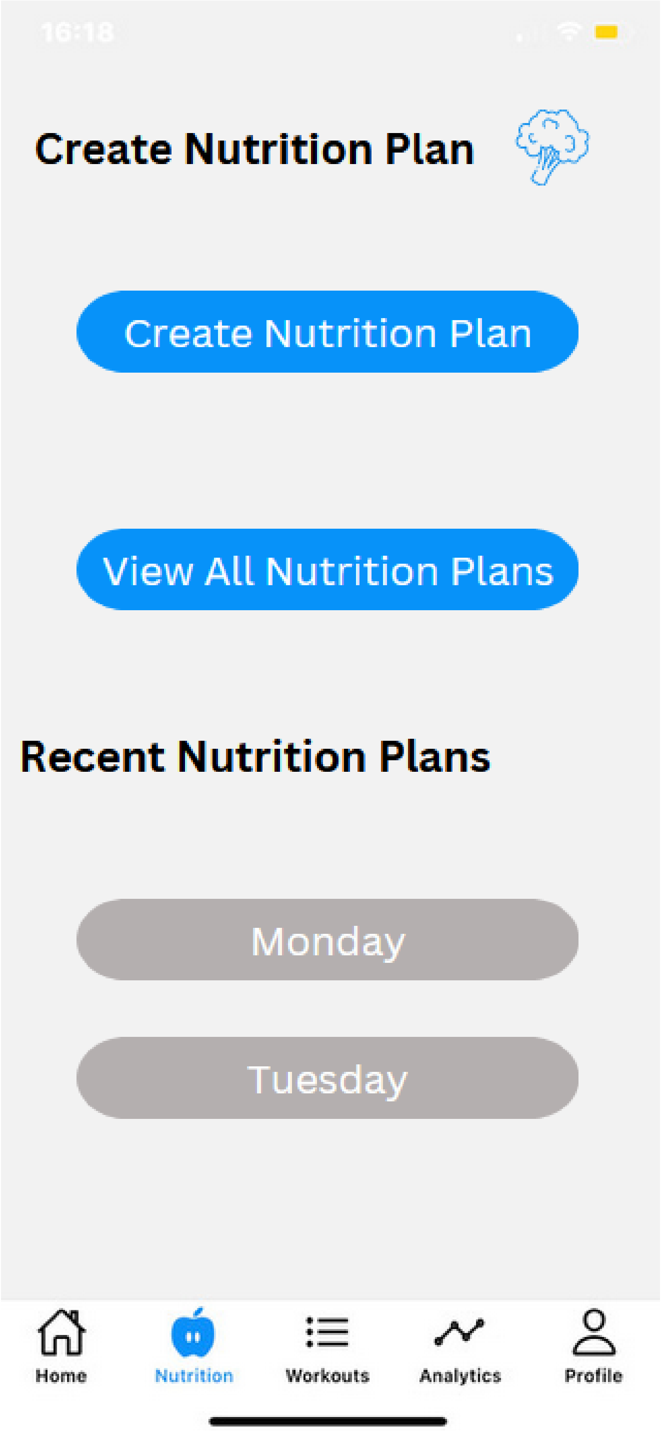


Figure 13 Prototype Nutrition Home Screen

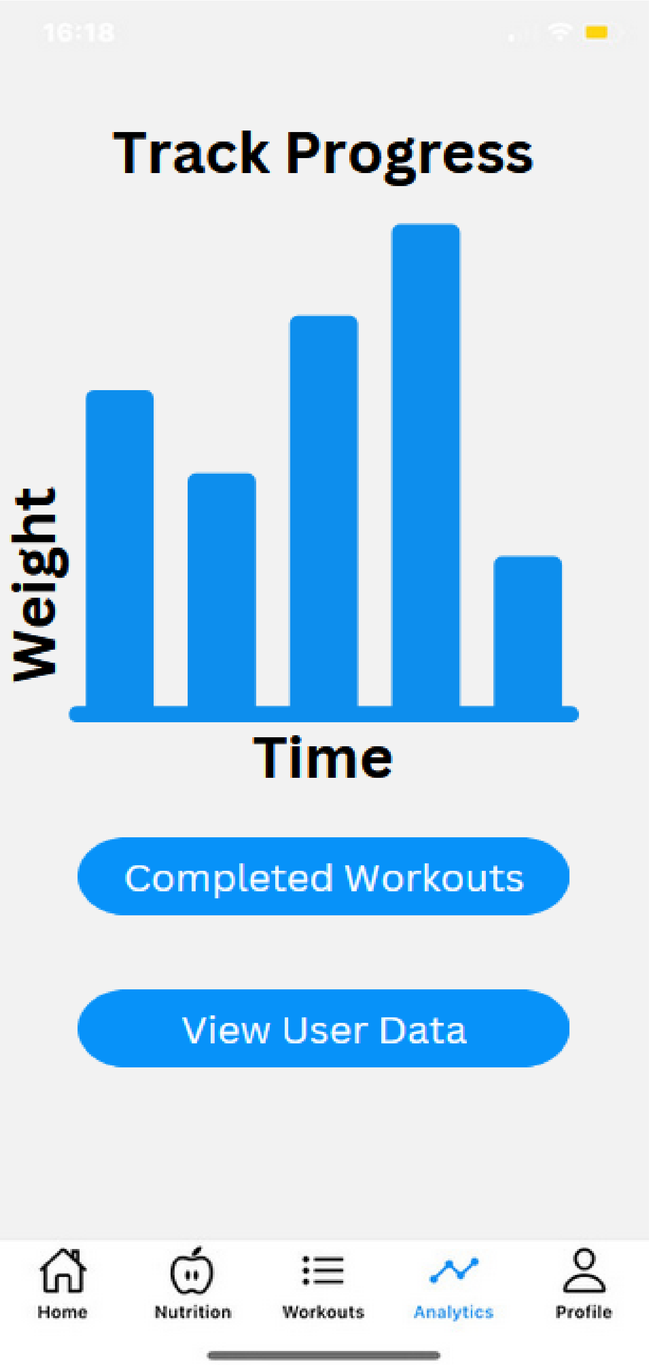
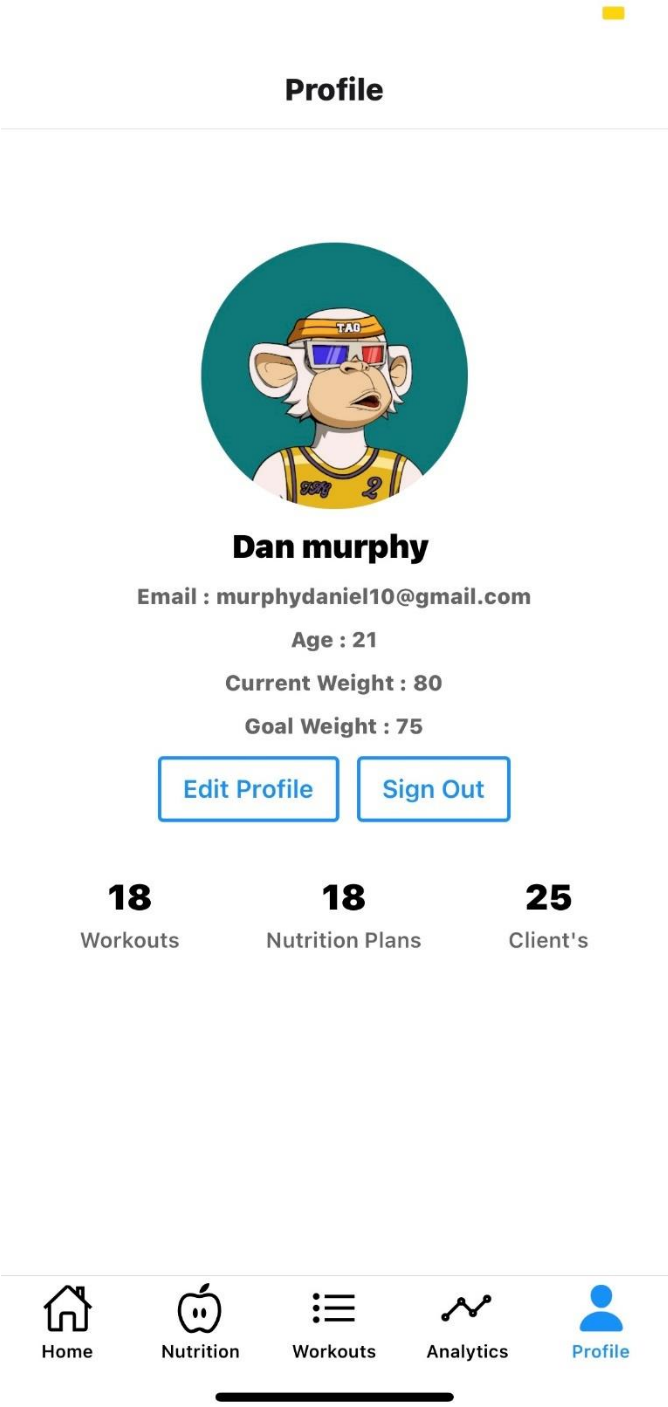


Figure 15 Prototype Analytics Screen



Dan murphy

Email : murphydaniel10@gmail.com

Age : 21

Current Weight : 80

Goal Weight : 75

18 Workouts

18 Nutrition Plans

25 Client's

Figure 16 Prototype Profile Screen

9. Bibliography

[1] Creately Blog. (2018). *The Ultimate Class Diagram Tutorial to Help Model Your Systems Easily*. [online] Available at: https://creately.com/blog/diagrams/class-diagram-tutorial/. [Accessed 13 Dec. 2022]

[2] Paterska, P. (n.d.). What is React Native and When to Use It For Your App? (Updated) - elpassion.com. [online]
Available at: https://www.elpassion.com/blog/what-is-react-native-and-when-to-use-it.
[Accessed 15 Dec. 2022]

[3] Educative: Interactive Courses for Software Developers. (n.d.). What is Firebase? [online]
Available at:
https://www.educative.io/answers/what-is-firebase. [Accessed 19 Dec. 2022]

[4] Sajjad, Z. (2021). *React Native Architecture: Creating Scalable RN Apps*. [online] Semaphore. Available at: https://semaphoreci.com/blog/react-native-architecture [Accessed 14 Jan. 2023].

[5] ActiveState. (n.d.). *What Is Pandas in Python? Everything You Need to Know*. [online] Available at: https://www.activestate.com/resources/quick-reads/what-is-pandas-in-python-everything-you-need-to-know/. [Accessed 16 Jan. 2022]

[6] [21] Guru99.com. (2019). What is TensorFlow? Introduction, Architecture & Example. [online] Available at: https://www.guru99.com/what-is-tensorflow.html. [Accessed 15. Jan. 2023]

[7] Firebase. (2019). *Choose a Database: Cloud Firestore or Realtime Database | Firebase*. [online] Available at: https://firebase.google.com/docs/database/rtdb-vs-firestore. [Accessed 16 Jan. 2023]