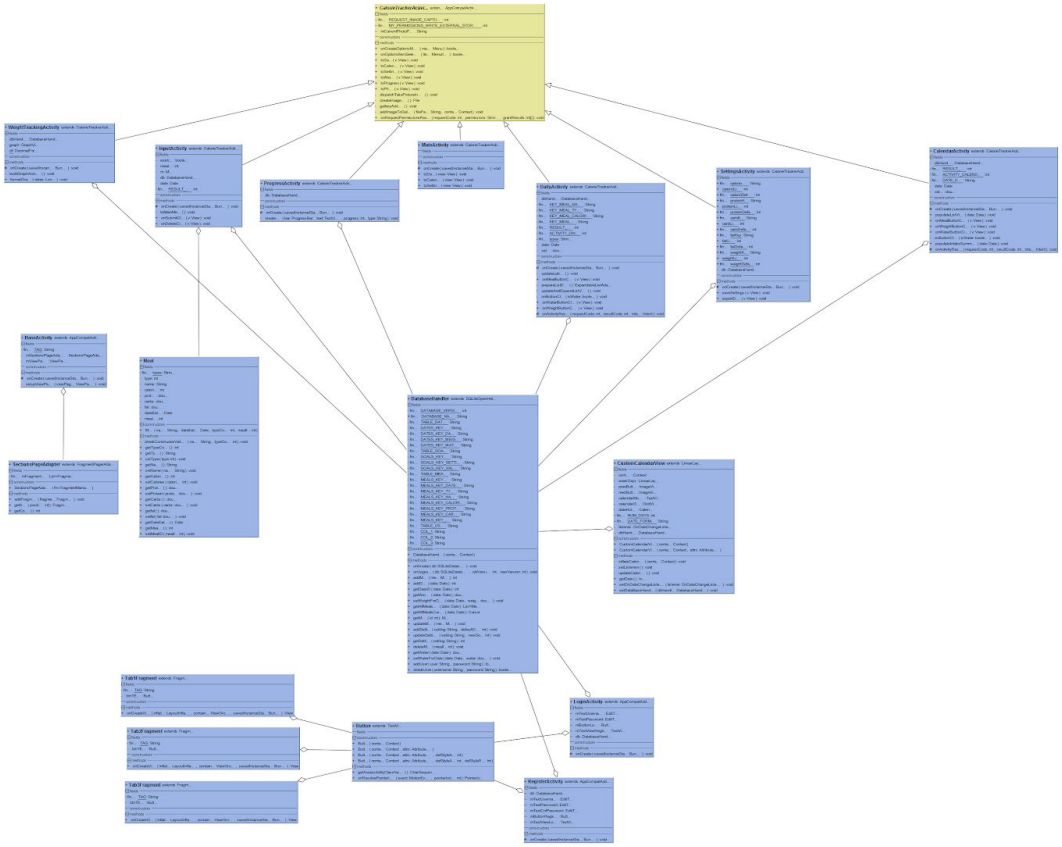
Capstone Part C

TJ Riker

Design Document

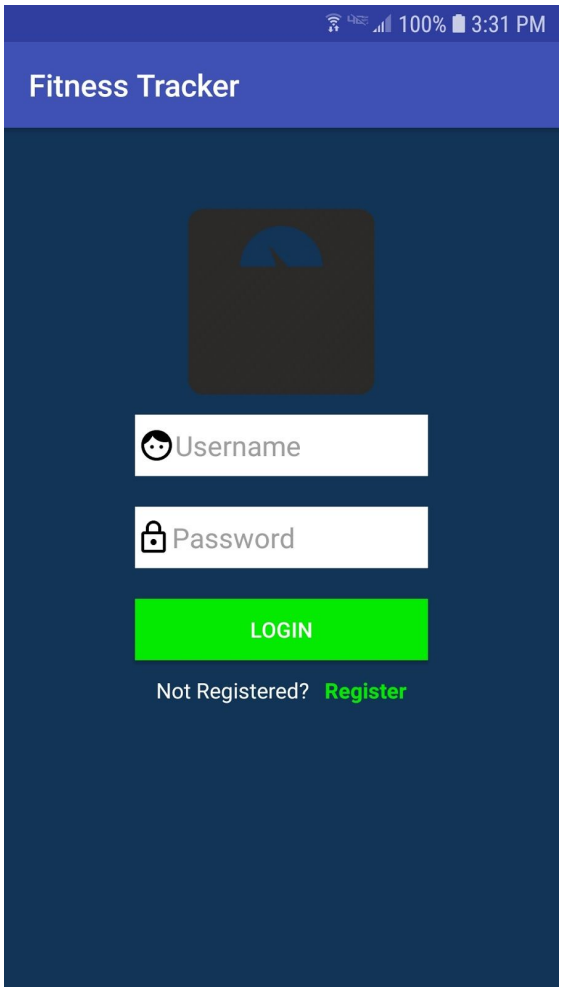
Class Diagram

Below is the Class Diagram UML of the entire project. This is included in the submissions for better viewing.

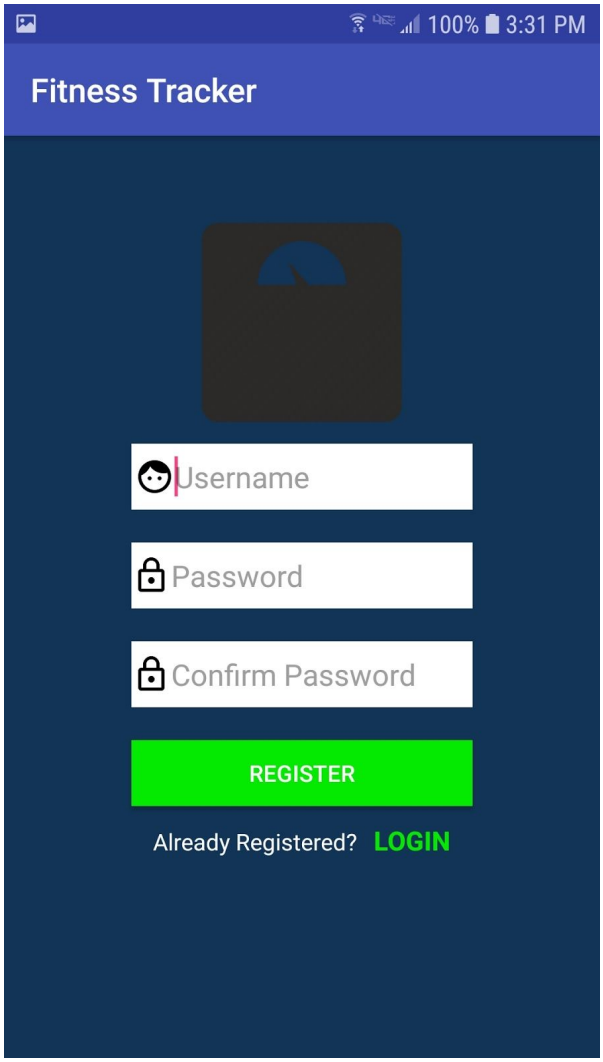


The “most important” activities are the Calorie Tracker Activity and the Database Handler, as many other activities are dependant on them. The top activity is the Calorie Tracker Activity. 7 other activities come off of the calorie tracker activity and are dependent on its values. From these 7 activities, 6 of them are connected to the Database Handler. The database handler also extends to 3 other activities that are dependent on the database functionality.

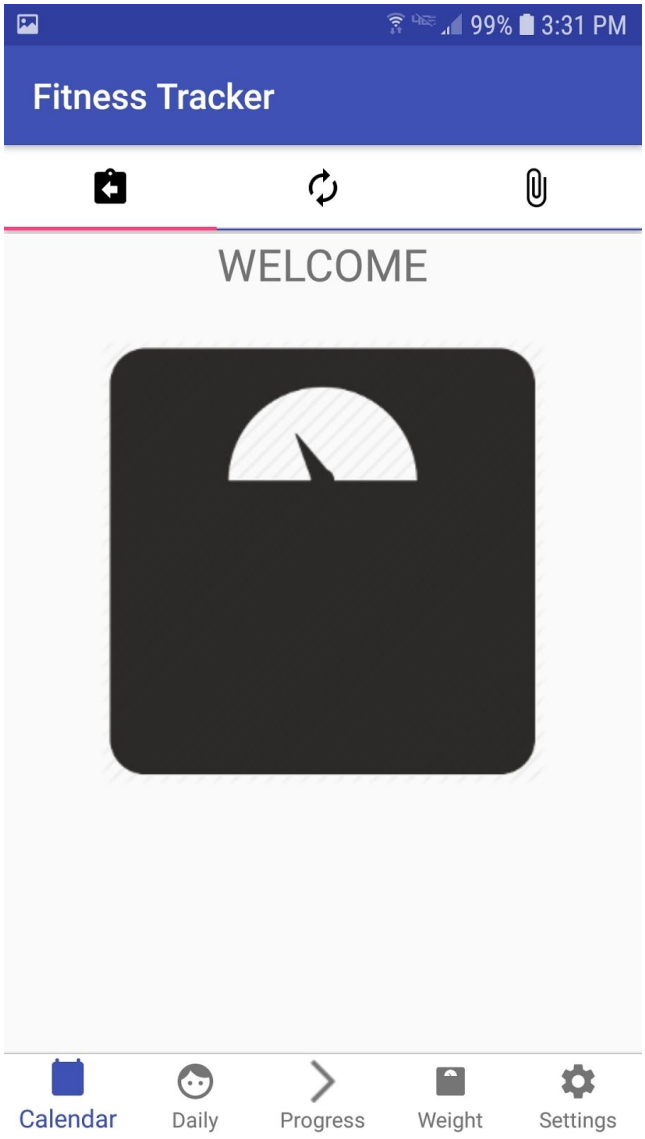
Design Diagram



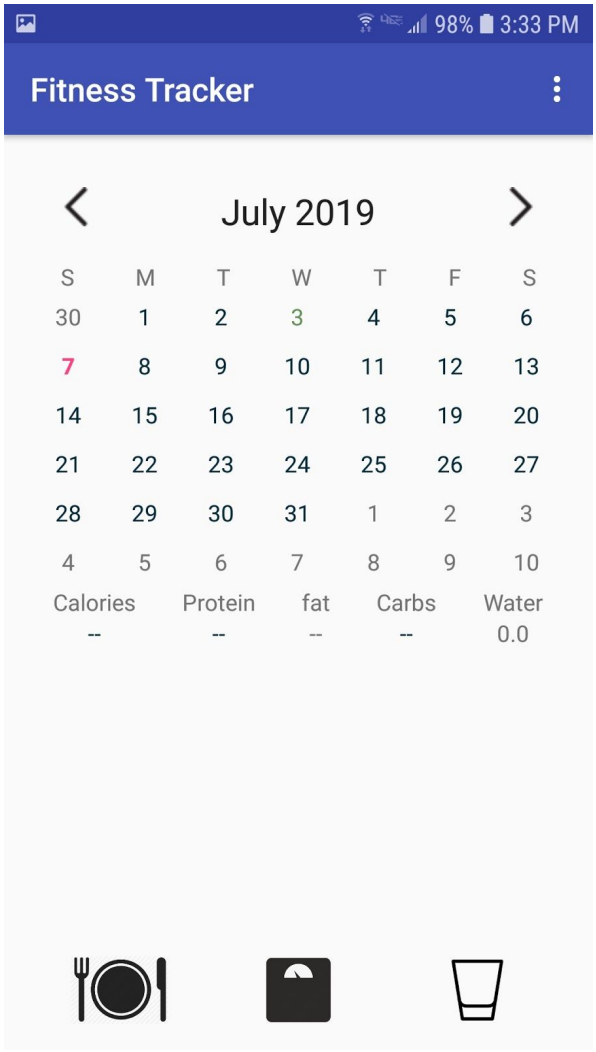
The very first screen upon opening the app is the “Login” screen. This is where, with an active account, the user would log into the full application. If the user does not have an account, they would click the green “Register” button to register for a new account.



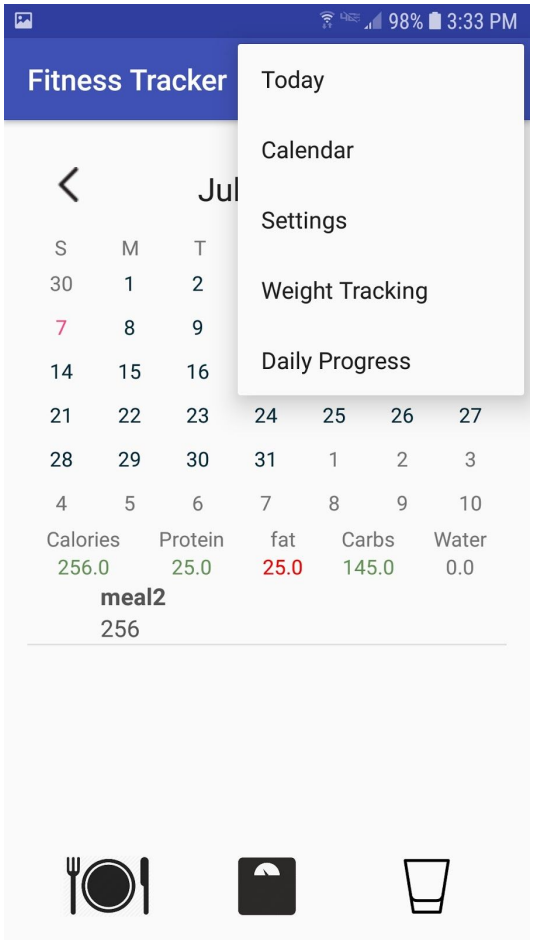
Upon clicking on the “register” button from the “login” screen, the user is taken to the “register” screen. This is the activity where the user would register for a new account. First, the user would need to input a username, and a password, followed by a confirmation of that password. Provided all inputs are correct, the user would then click on the “Register” button, and a new account would be created. The user would then be taken back to the previous “Login” screen.



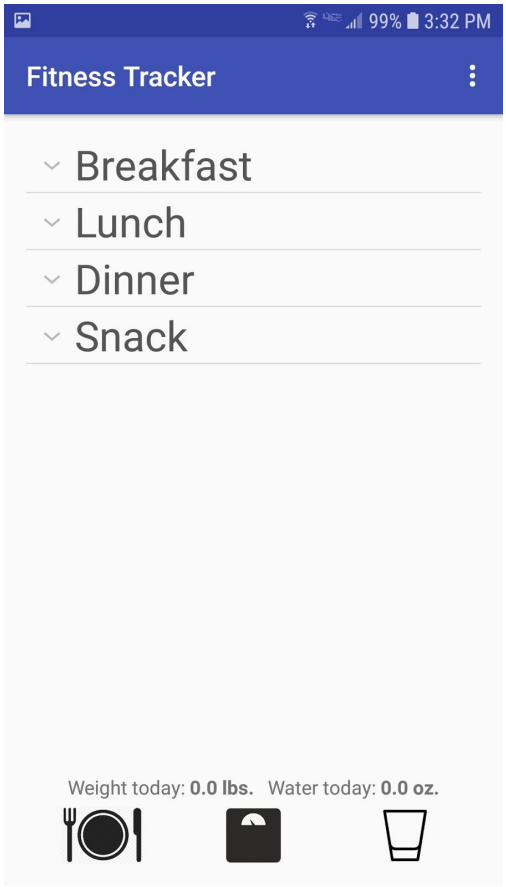
Once logged in, the user would be taken to this screen, the main menu. From here, there are many different options. We will go through them in order for the purpose of this explanation. The first button is the “Calendar” button.



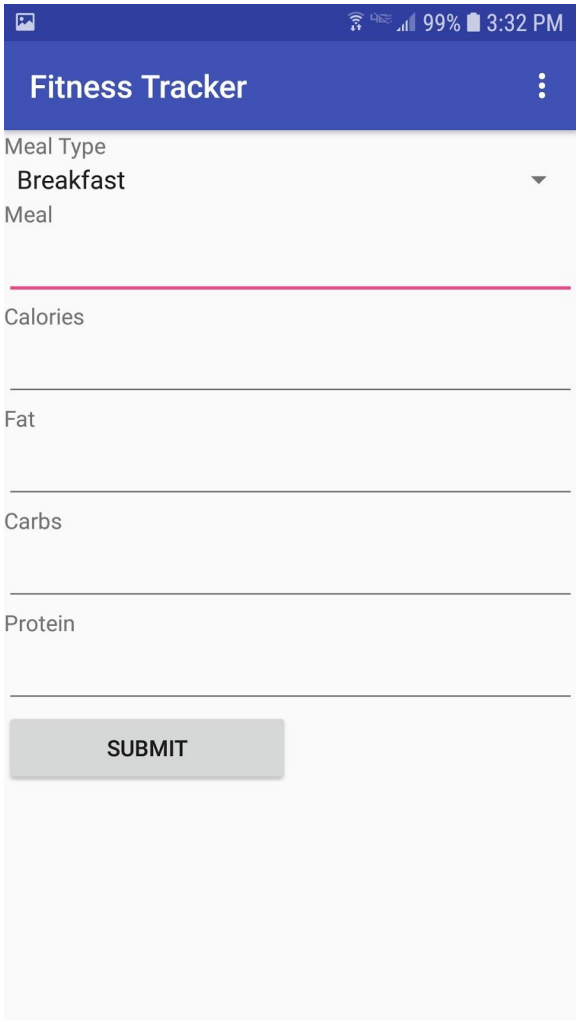
Upon pressing the “Calendar” button, the user is taken to this screen. This is the Calendar activity, where a user can see all of their statistics for any given day. Clicking on a day in the calendar will show the statistics for that specific day.



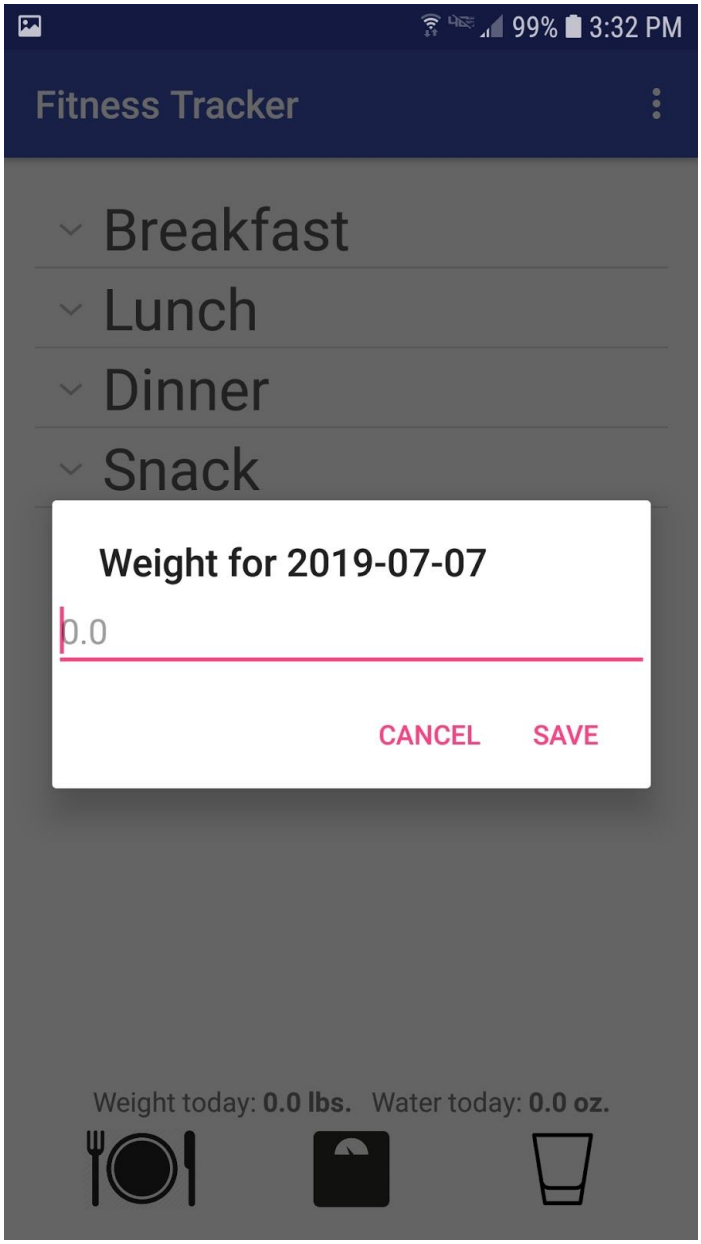
From any activity, the user can press the “back” button to be taken back to the “Main Menu”. Alternatively, the user can press the drop down menu in the top right corner of any activity, for a quick view at the “Main Menu” activities, as shown here.



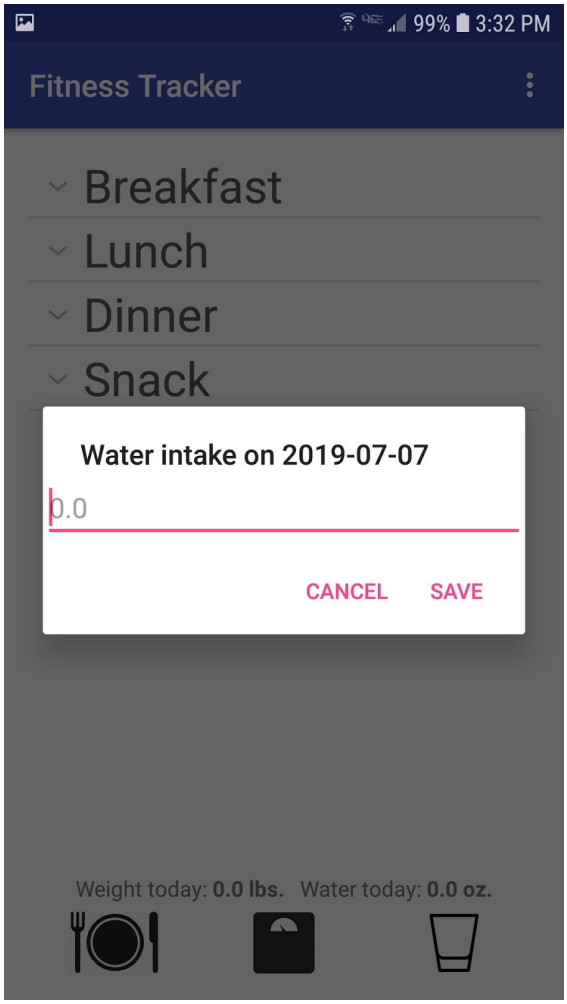
The next activity is the “Daily” activity, and most likely where a user would be spending most of their time. From this activity, a user can add meals to their daily plan, seperated into either Breakfast, Lunch, Dinner, or Snacks. The user can also add their weight for that day by clicking on the “scale” icon. Clicking on the “water glass” icon will bring up a prompt for the user to enter the water that they have consumed.



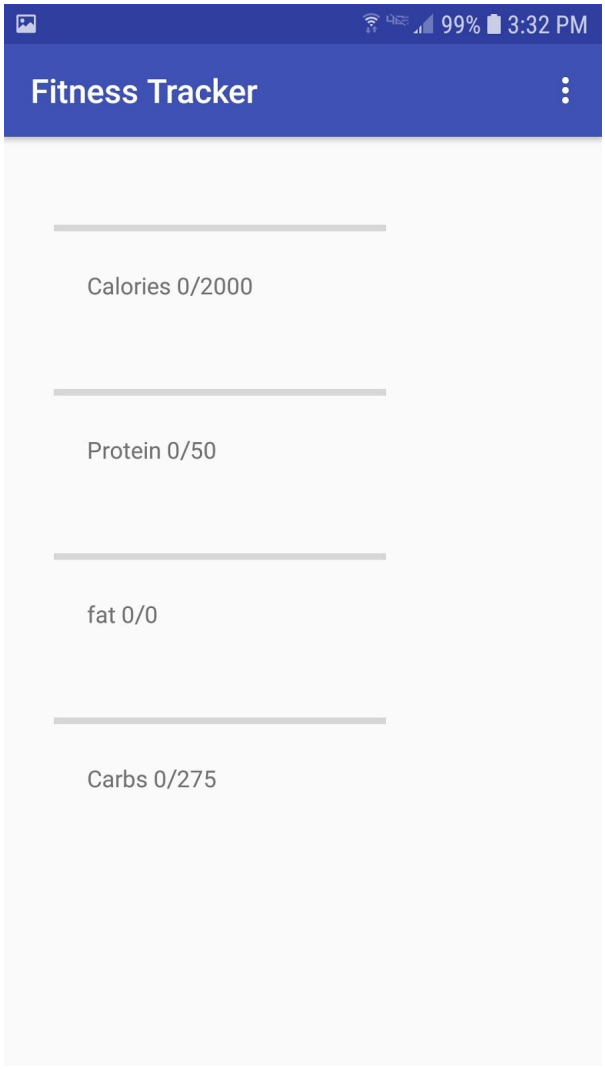
Adding a meal.



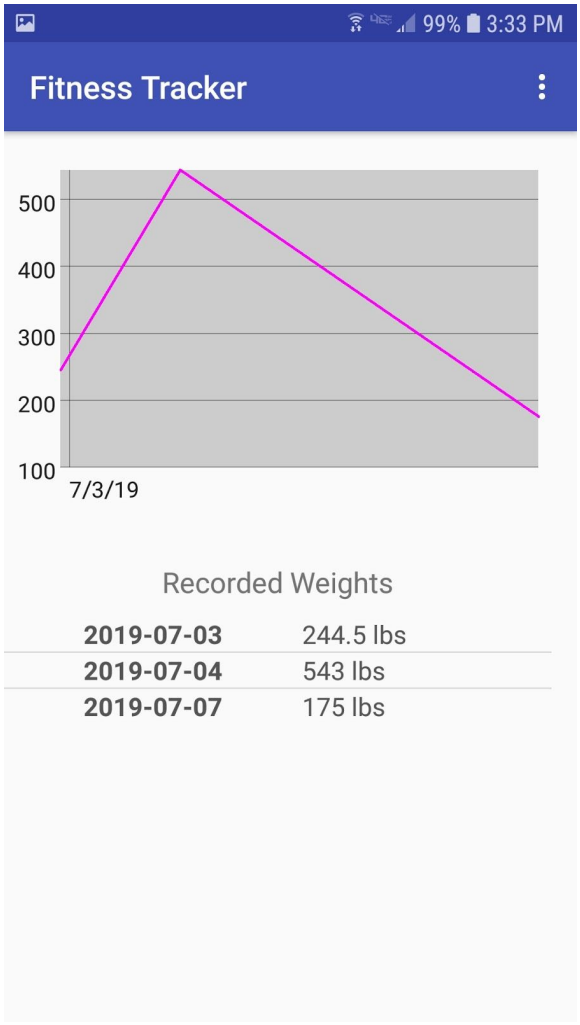
Entering weight for a specific day.



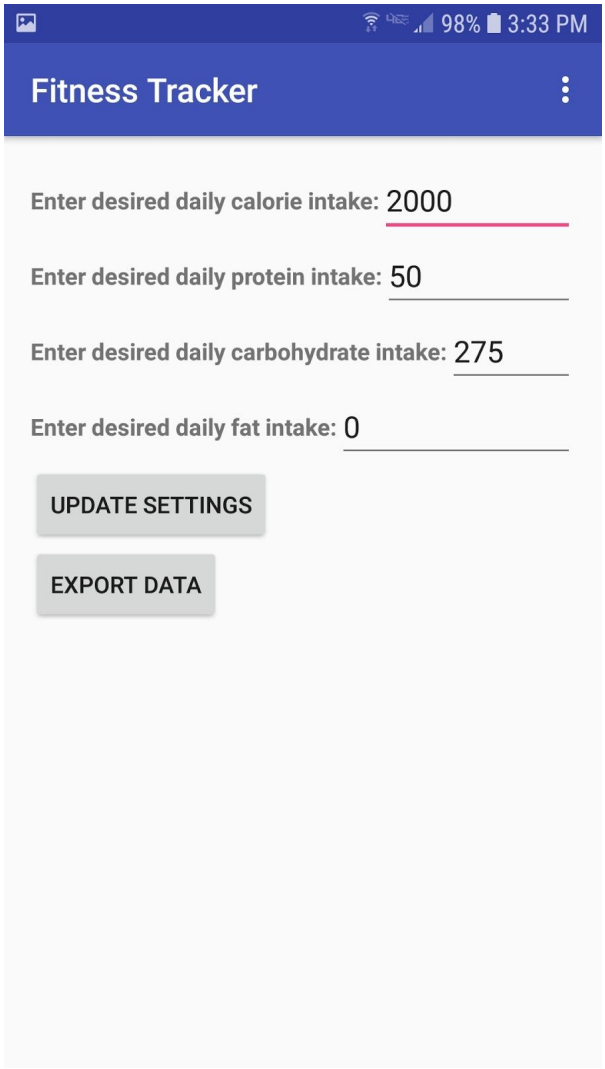
` Adding water intake.



The next activity is the “Progress” activity. This view gives a look at the progress made so far in that specific day. These values are based on the values provided in the “Settings” activity.



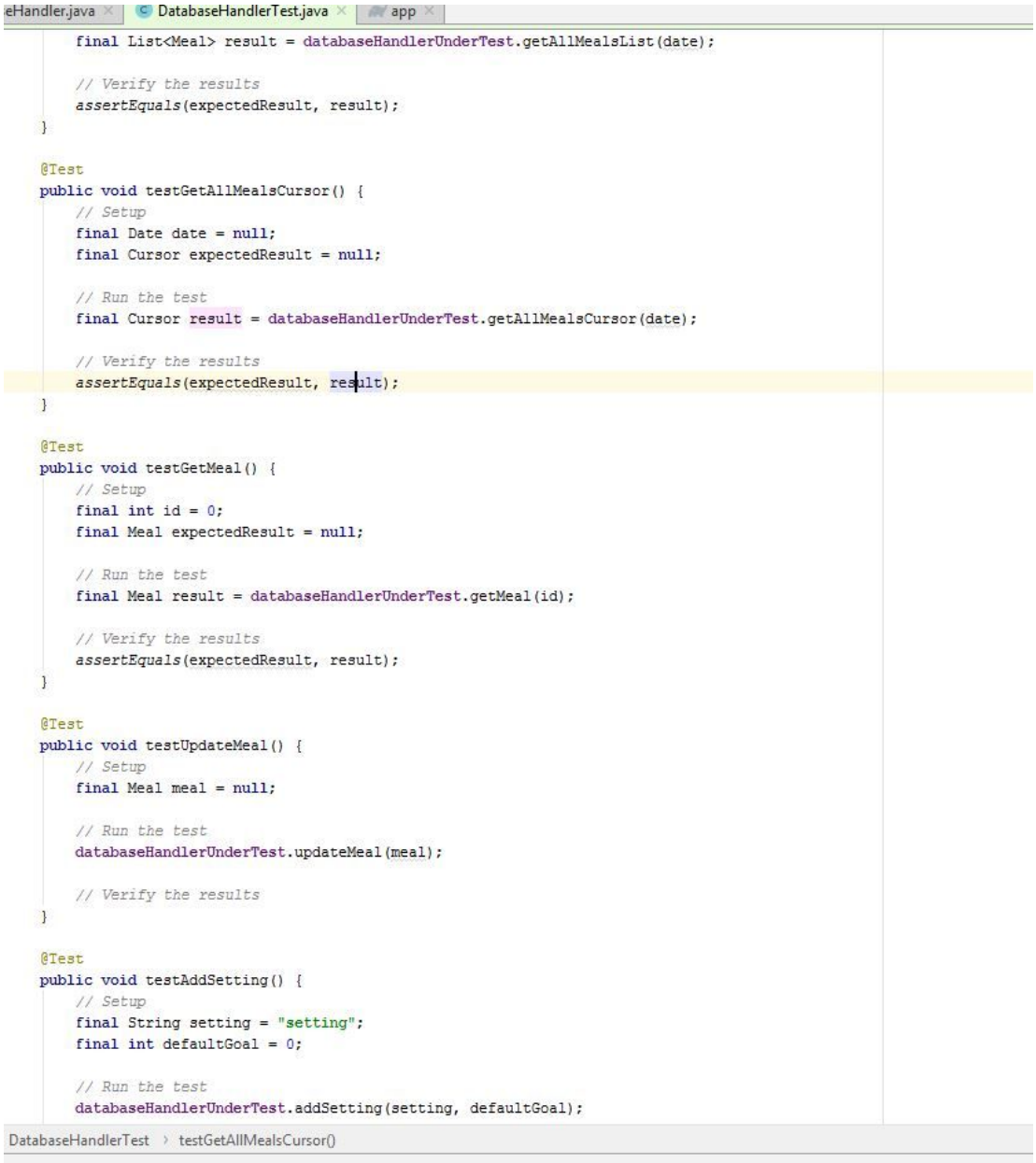
The next activity is the “Weight” activity, which gives a list and graph view for the weights that have been recorded.

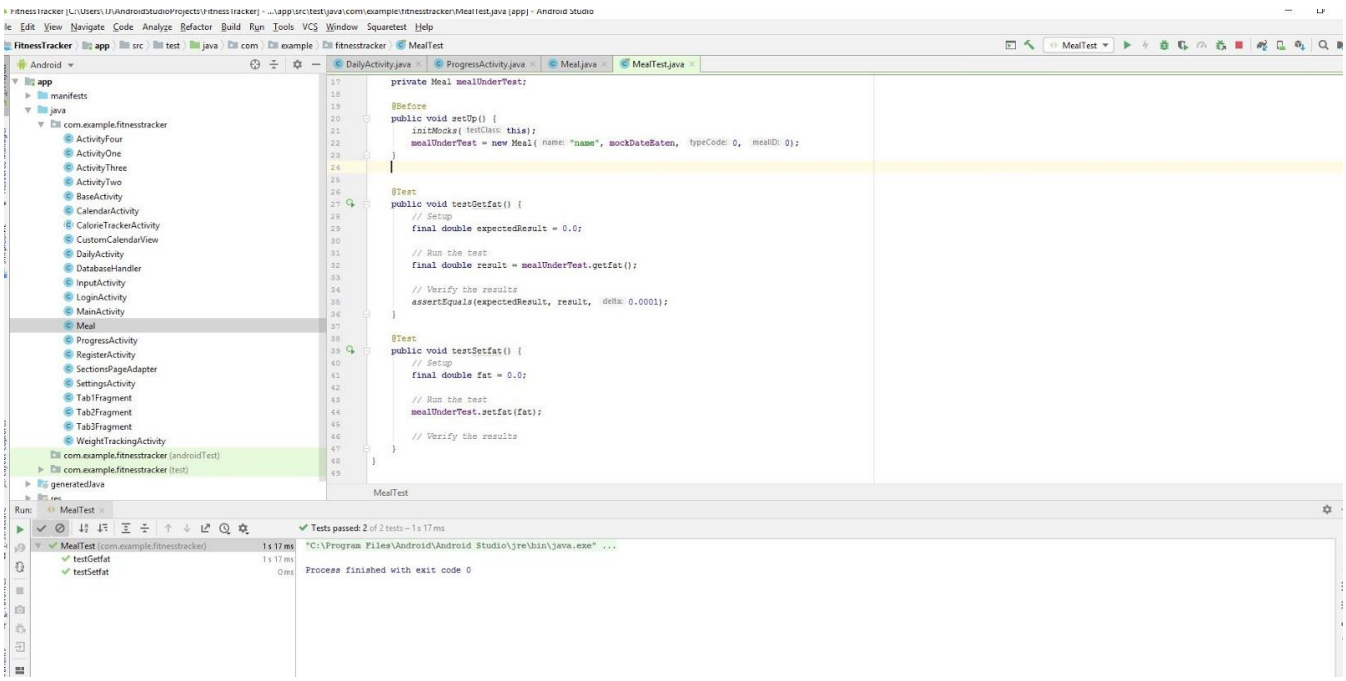


The final activity is the “Settings” activity. This is where a user inputs their desired goals for calorie and macronutrient intake. These values can be changed at any time and will be dynamically updated throughout the entire app. The user can also choose to export the contents of their personal database for viewing, if they wish.

Unit Tests

This application was heavily dependant on a fully functioning SQLite Database. The unit testing for this app was mainly done on the database functionality, ensuring that it is working correctly. Most tests were done in a fashion of inputting data into the database, and then retrieving that data, and making sure it was exactly as it was expected to be.





The testing was done with JUnit4 and Mockito. Testing a database turned out to be quite a challenge and I had to do a lot of research on learning how to properly test a database. The testing was very helpful though and showed where there were bugs, allowed me to pinpoint those bugs, and fix them, before moving onto a different area. Testing was done throughout the process of developing the application and I believe it really helped keep bugs and other problems to a minimum, as when a new bug was introduced, I would quickly find out about it during the next testing run.

Source Code/Executable File

The source code for this application is included in the submission. The source code can be found in the FitnessTracker.zip. Unzip this folder. Navigate to app/src/main/java/com/example/fitnesstracker. From here, all of the source code can be seen.

The executable file is in the form of an APK. This can be found in the ExecutableAPK.zip. Unzip this folder and it is the app-release.apk. The keystore is also found in this folder.

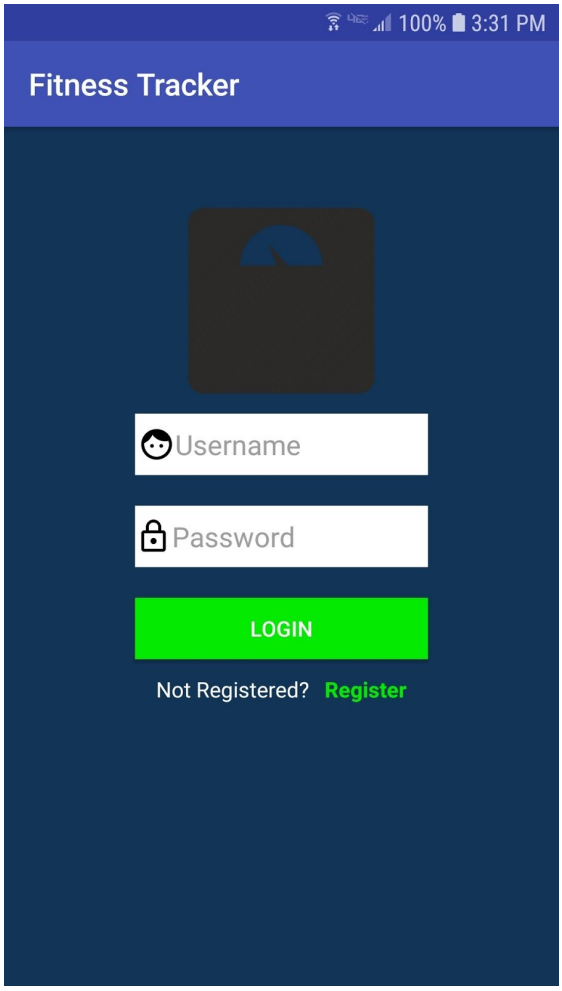
User Guide

Maintenance Purposes

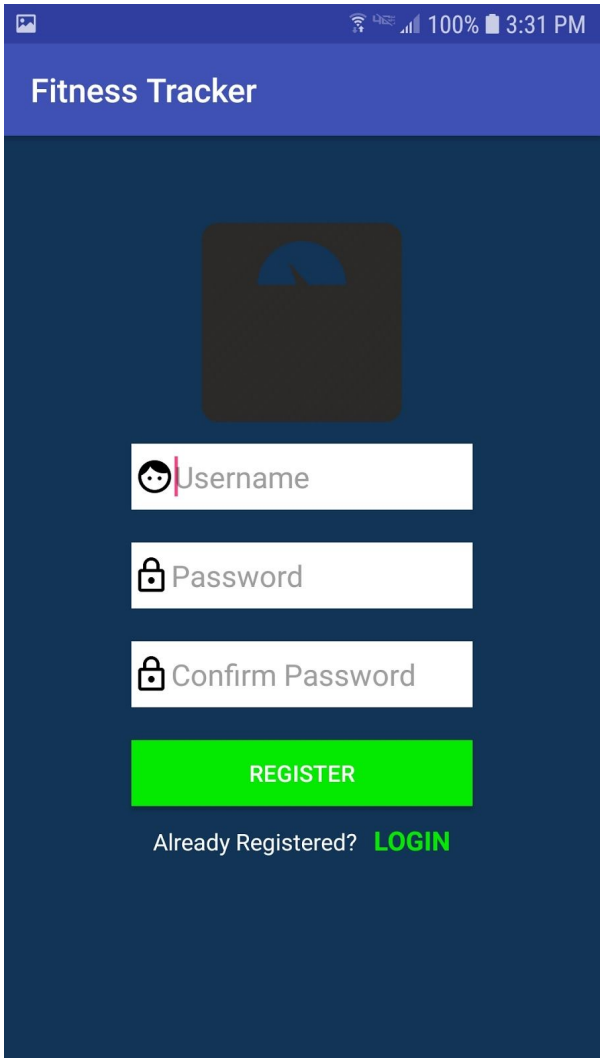
For maintenance purposes, the user doing the maintenance or testing would need to directly install the apk to the android device that they want to test on. If a maintainer wishes to alter the code for maintenance purposes, they would need to install the package into android studio. From here, the source code and other files could be altered. Then, the user would need to connect and android device to their PC through a USB cable. The USB Debugging feature on the device would need to be turned on. Then the APK could be directly installed to the device through android studio.

User Perspective

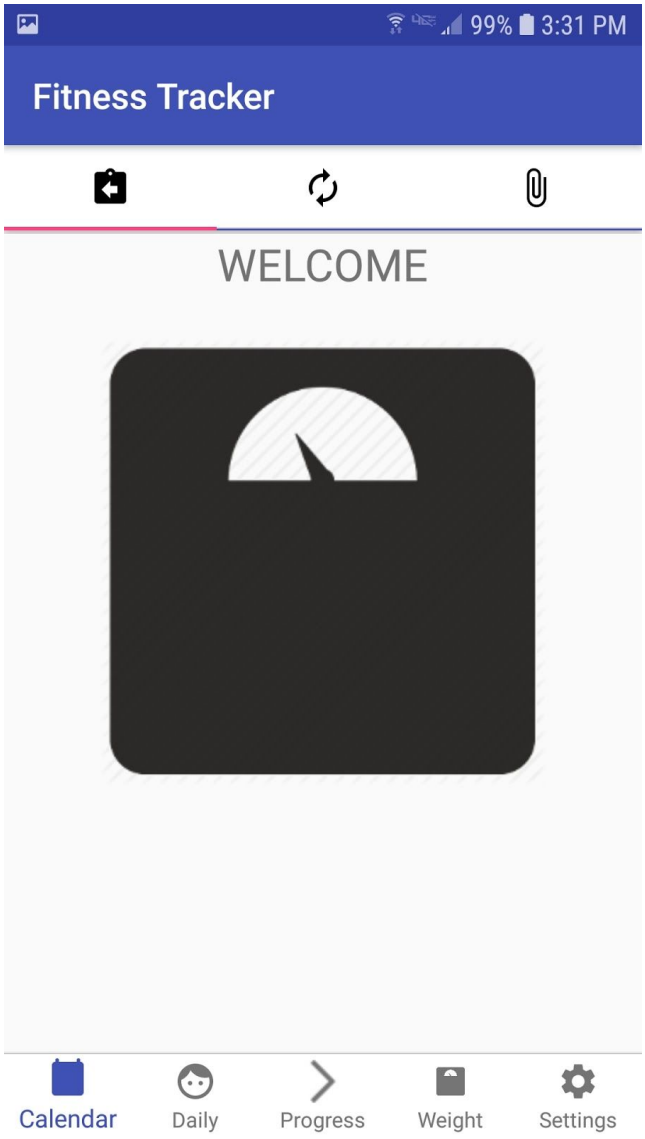
The perspective of an end user trying to use this application is much easier. When the application is hosted to the Google Play Store, anyone with an android device can easily download the app, directly to their phone. From here, they just need to open the application. Doing so will take the user to the “Login” screen.



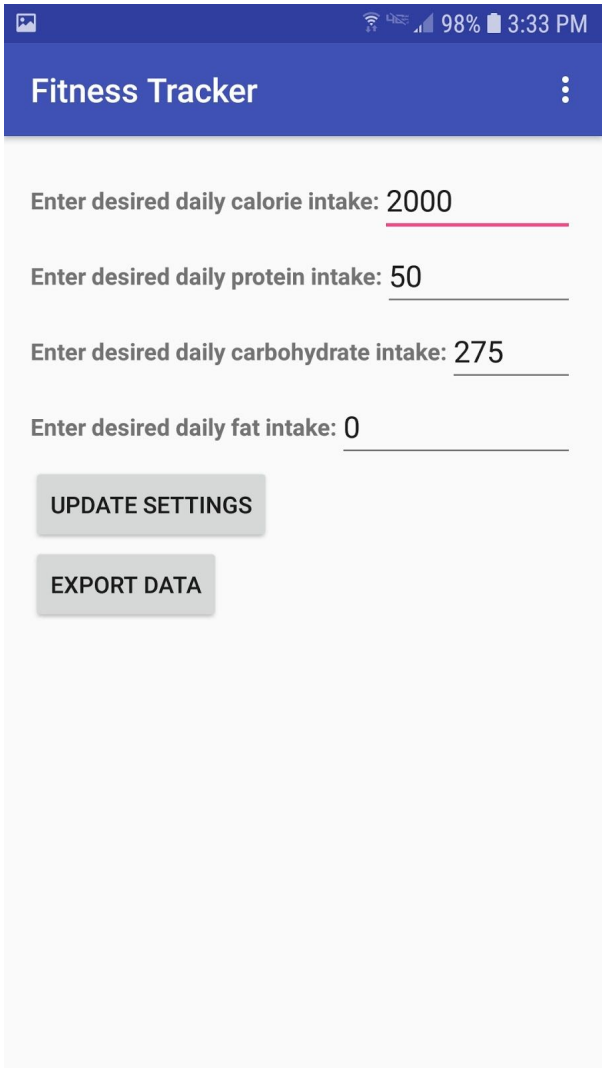
The very first screen upon opening the app is the “Login” screen. This is where, with an active account, the user would log into the full application. If the user does not have an account, they would click the green “Register” button to register for a new account.



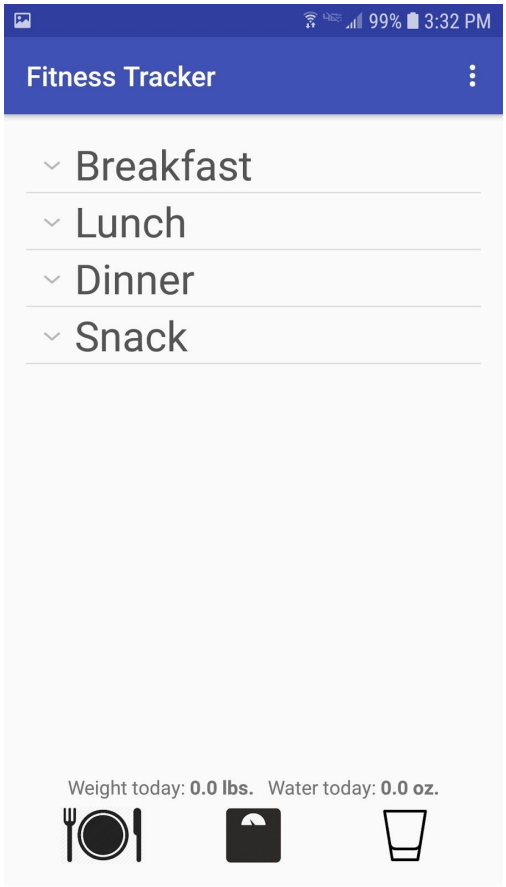
Upon clicking on the “register” button from the “login” screen, the user is taken to the “register” screen. This is the activity where the user would register for a new account. First, the user would need to input a username, and a password, followed by a confirmation of that password. Provided all inputs are correct, the user would then click on the “Register” button, and a new account would be created. The user would then be taken back to the previous “Login” screen.



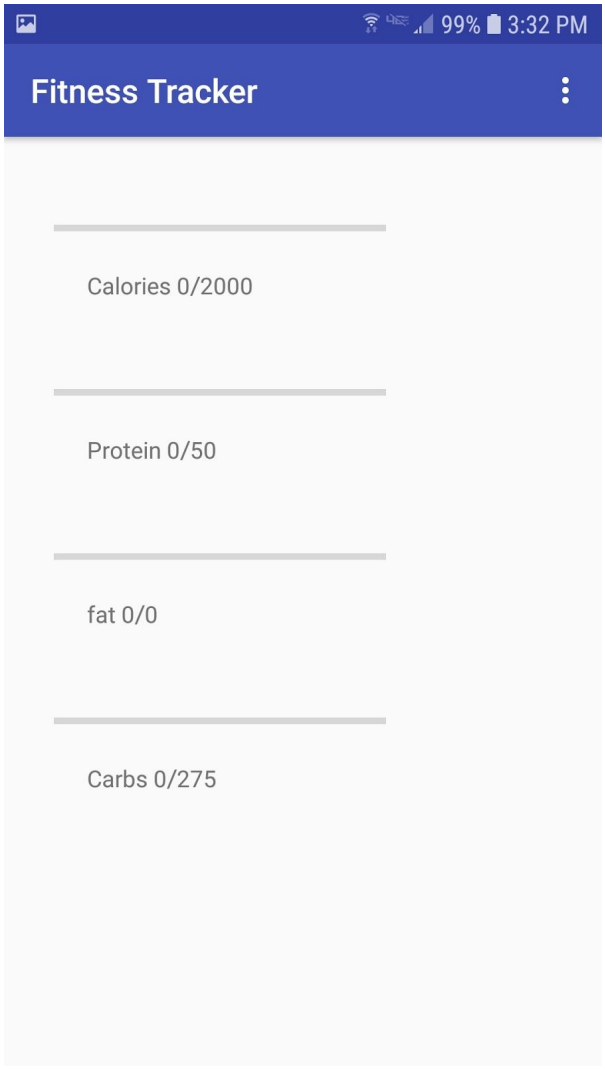
Once logged in, the user would be taken to this screen, the main menu. From here, to set up the application for a first time use, I recommend clicking on the “Settings” tab.



The user would then be taken to this screen where they can enter their desired goals. This is all that needs to be done to customize the experience to a specific user. But as an explanation for a new user, I would guide them to the “Daily” activity next.



The end user, with their customized values from the “Settings” activity, can now start adding meals and other information into the app from this page. The user can then look at the “Progress” activity next if they wish to see the statistics they are entering being compared to their overall daily goals.



That is all that needs to be done to get a basic grasp on the application and set it up for a specific end user's goals.