GitHub Username: ferdinandtongson

Gym Rat, Personal Trainers' Assistant

Description

Gym Rat, Personal Trainers' Assistant is the ultimate Assistant for Personal Trainers. It provides the tools you needed to manage appointments, organize client records, assist in training sessions and remember your favorite exercises and routines so you can focus on what you do best, inspiring and motivating your clients to meet their fitness goals.

Gym Rat manages your Appointments

- View all your appointments by the day or week
- Easily add or remove appointments and record cancellations
- Quickly contact clients by phone, sms or email

Gym Rat organizes your Client Records

- Keep track of all of your clients info and goals
- Access previous training sessions, body measurement stats, and SOAP note records
- Schedule future appointment with the option to pre-define the type of exercise routine that will be used for that session

Gym Rat assists in your Training Sessions

- Record body measurement with calculated change results to give for immediate feedback
- Track each exercise, weight and rep or time and distance in your training session
- Use pre-defined set of exercises for the session or add exercises on the fly during
- Use SOAP notes to keep track of how your client is doing at the end of the session.

Gym Rat remembers your Exercises and Routines

- Save your favorite exercises to a list, separated by category
- Create training session routines by pulling from your list of favorite exercises

Intended User

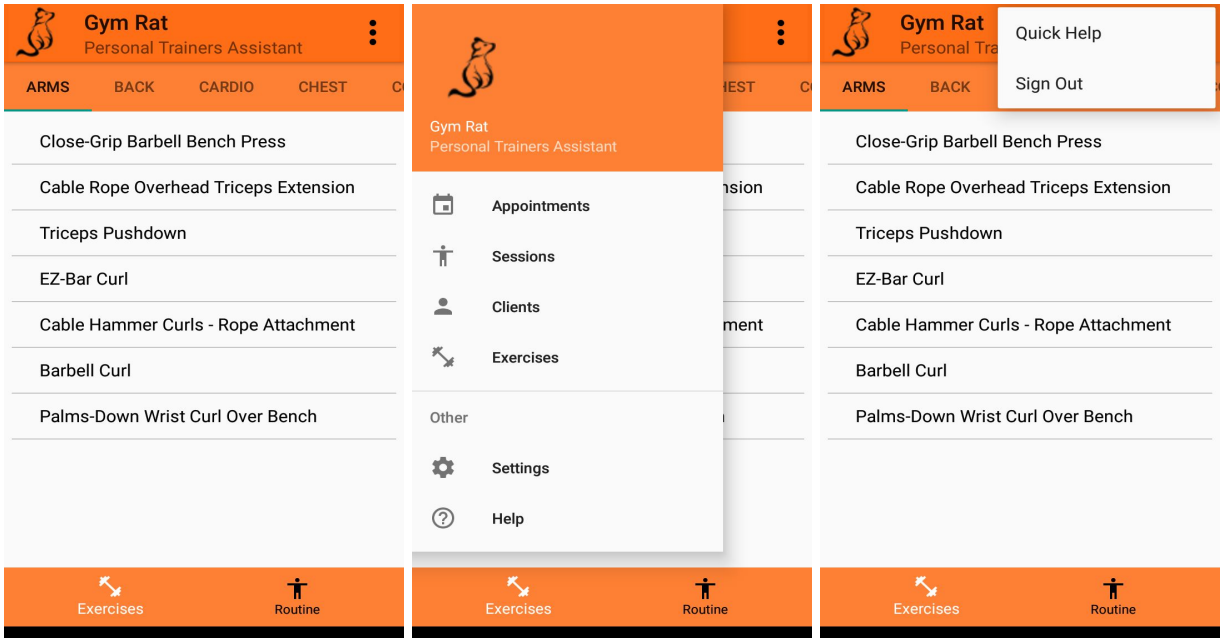
This app is for Personal Trainers managing a list of clients

Features

- Saves information
    - Manages appointment info
    - Client record info
    - Training session inf
    - Exercise and Routine info
- Send notifications to you and your client for upcoming training sessions
- Sync data to all devices linked to the users login account

User Interface Mocks

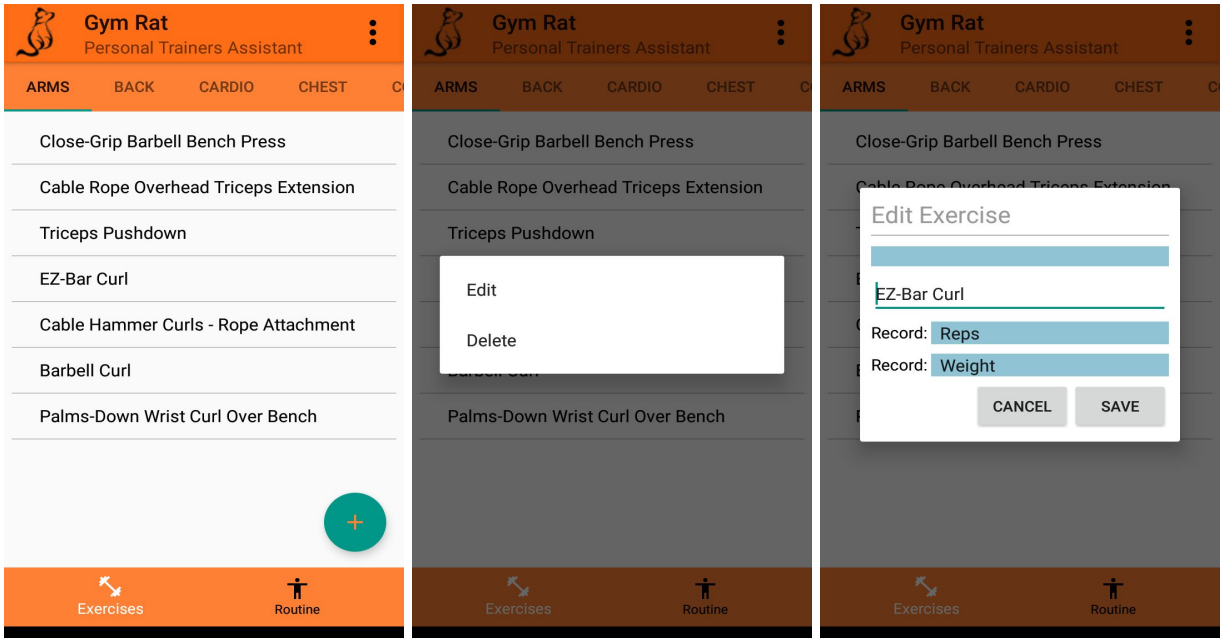
Main Navigation



The Toolbar component contains a Drawer component through its' Navigation Icon and Quick Help and Sign Out options through its' Options Menu.

The Drawer component will give access to Appointments, Sessions, Clients and Exercise modules as well as Settings and Help dialogs.

Exercise Screen



The Exercise screen is where the user can create a list of his/her favorite exercises separated by category. When creating an exercise, the user has the option to save up to 2 types of records linked to that exercise. In the case above, the EZ-Bar Curl exercise will record the number of reps and the weight used during the exercise.

The Exercise screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, ViewPager, RecyclerView, and a Floating Action Button component.

The Bottom Navigation allows the user to switch from the Exercise screen to the Routine screen.

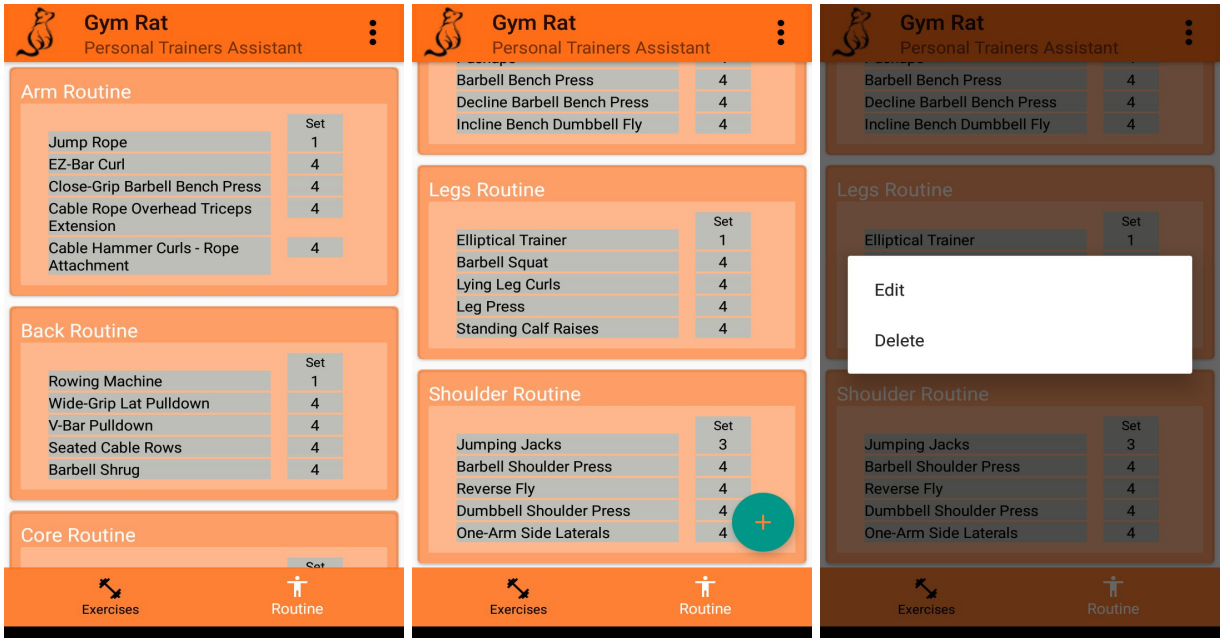
The ViewPager with TabLayout allows the user to scroll through various lists of exercises separated by user-defined categories.

The RecyclerView displays the list of user-defined exercises by category. Long-clicking on a list item will open a context menu that will give the user the option to edit or delete the item.

The FAB component allows the user to add a new exercise item to a given list. The FAB will only be displayed when the user flings-up at the bottom of the list of items.

Dialogs are used to edit or add new exercises to a list.

Routine Screen



The Routine screen is where the user can see the list of routines he/she has predefined. Each routine item will display the list of exercises to be done in the routine along with the number of sets to be done in each exercise.

The Routine screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, RecyclerView, and a Floating Action Button component.

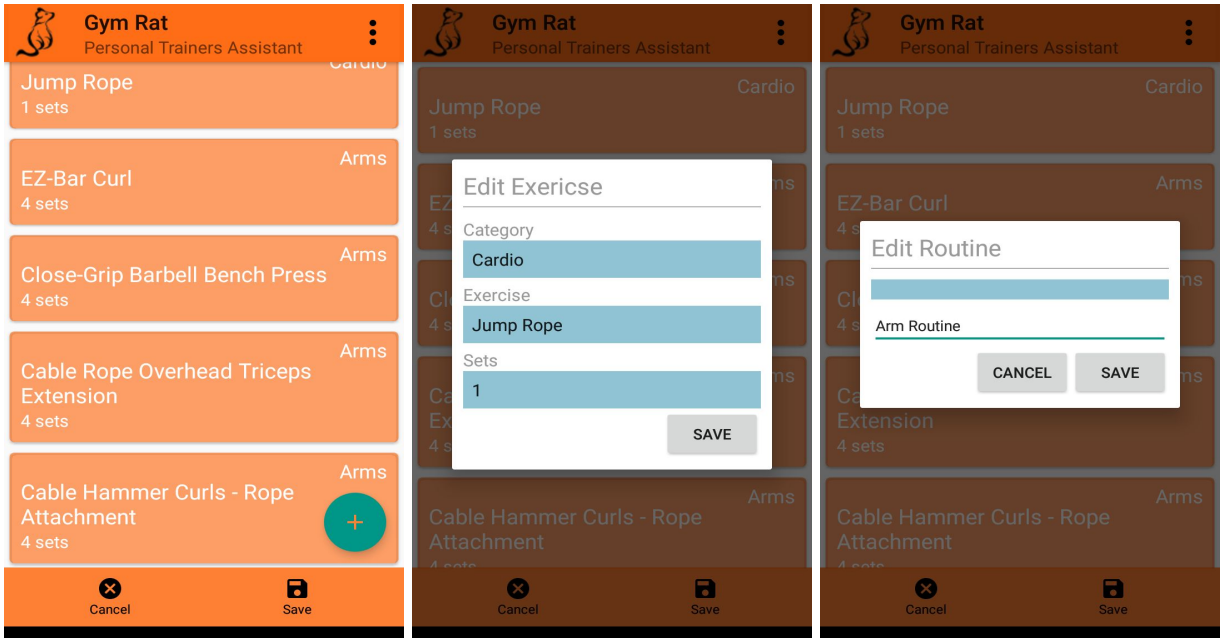
The Bottom Navigation allows the user to switch from the Routine screen to the Exercise screen.

The RecyclerView displays the list of user-defined exercise routines. Long-clicking on a list item will open a context menu that will give the user the option to edit or delete the item.

The FAB component allows the user to add a new exercise routine item to the list. The FAB will only be displayed when the user flings-up at the bottom of the list of items.

Adding or editing a exercise routine will take the user to the Edit Routine Screen.

Edit Routine Screen



The Edit Routine screen is where the user can create exercise routines. A list of exercises used in the routine will be displayed in order, if any. Adding or editing an exercise item will give the user the option to select the exercise category, exercises within the category and number of sets to be done in the exercise.

The Edit Routine screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, RecyclerView, and a Floating Action Button component.

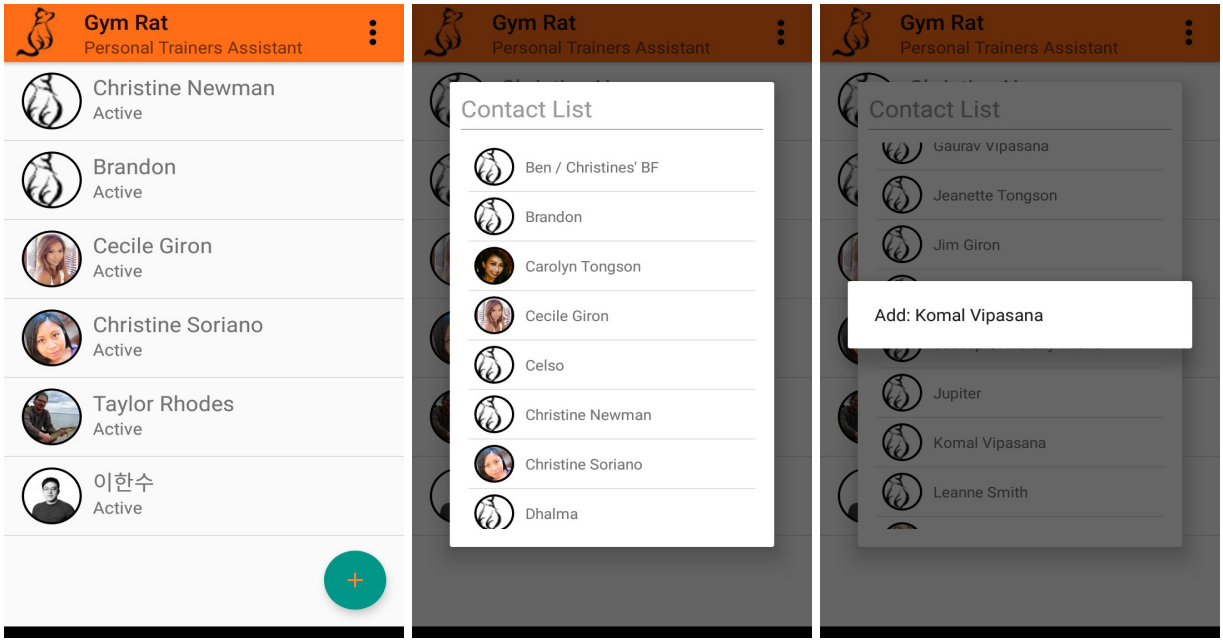
The Bottom Navigation allows the user the option of saving the routine or canceling out of the screen. If the routine is to be saved, a dialog will appear for the user to input the routine name

The RecyclerView displays the list of user-defined exercise routines. Long-clicking on a list item will open a context menu that will give the user the option to edit or delete the exercise item.

The FAB component allows the user to add a new exercise item to the routine. The FAB will only be displayed when the user flings-up at the bottom of the list of items.

Adding or editing an exercise item in the routine will open a popup dialog. The dialog will contain Spinner options for exercise categories, exercises within the selected category and number of sets to be done during the exercise routine.

Clients Screen



The Clients screen is where the user can see his/her list of clients and can access more detailed information about a particular client by clicking on a list item. The user can also add clients from his/her list of contacts contained in the phone or tablet.

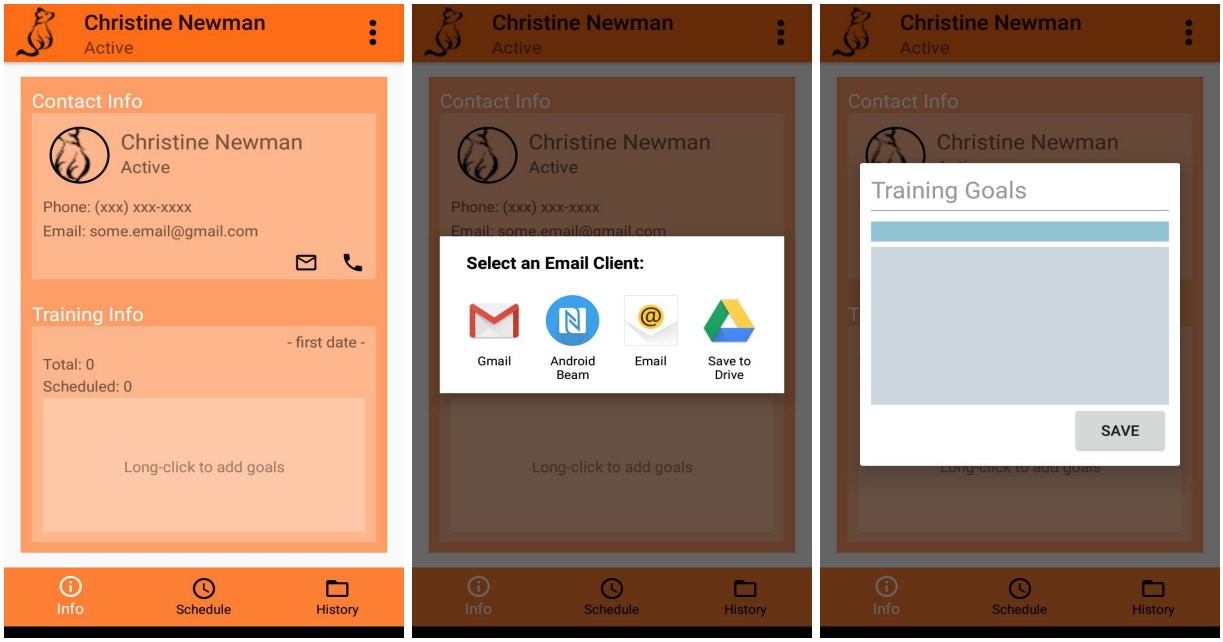
The Clients screen contains a Toolbar (see Main Navigation for more info), RecyclerView, and a Floating Action Button component.

The RecyclerView displays the list of user clients. Clicking on client will take the user to the Client Info screen.

The FAB component allows the user to add new clients to the list. The FAB will only be displayed when the user flings-up at the bottom of the list of items.

Adding a client will open a popup dialog. The dialog will display a list of all of the user's contacts stored on the phone. Long-clicking on any one of the contacts on the list will open a context menu to confirm the addition of the contact to the client list or to notify the user that the contact has already been added as a client.

Client Info Screen



The Client Info screen is where the user can find more information about the client's phone number, email, first training session date, training session total, number of future sessions schedule, and client's training goals. The user can also send the client an email, sms message or make a phone call from this screen.

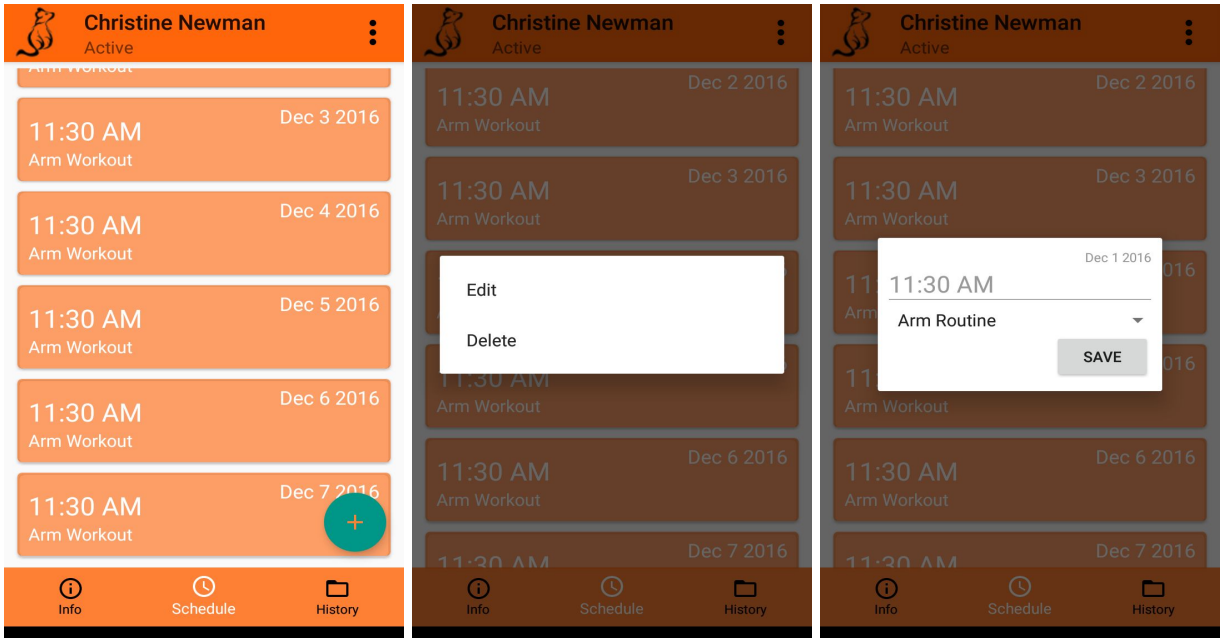
The Client Info screen contains a Toolbar (see Main Navigation for more info) and Bottom Navigation component.

The Bottom Navigation allows the user to switch between the Client Info, Client Schedule and Client History screens

There is an email icon, that will allow the user to send an email to the client, and a phone icon, that will allow the user to send an sms message or directly call the client.

Long-clicking on the "goals" textView component will bring bring up a dialog for the user to input the client's training goals.

Client Schedule Screen



The Client Schedule screen is where the user can schedule future training sessions with the client.

The Client Info screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, RecyclerView and FAB component.

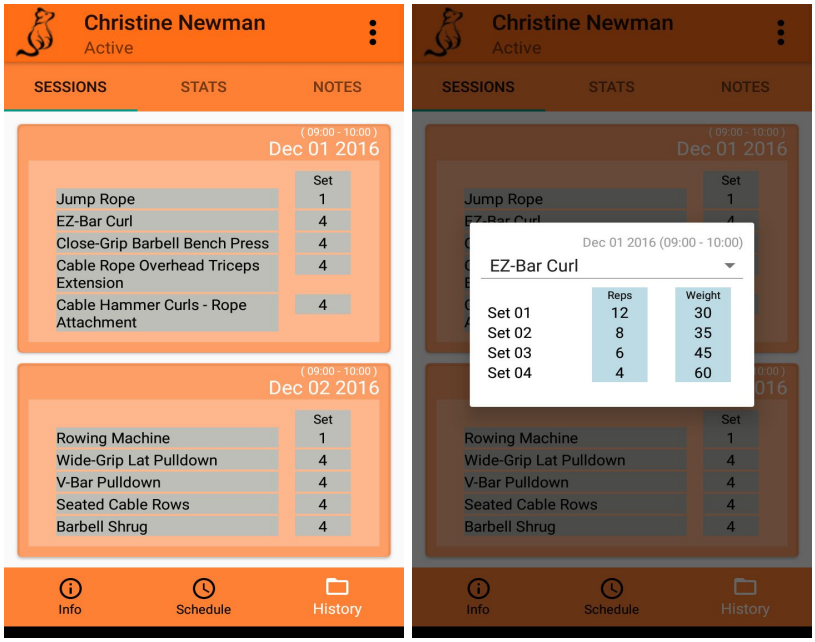
The Bottom Navigation allows the user to switch between the Client Info, Client Schedule and Client History screens

The RecyclerView displays a list of future training sessions with the client and the type of training routine scheduled for that session, if any. Long-clicking on a list item will open a context menu that will give the user the option to edit or delete the item.

The FAB component allows the user to add a new schedule to the list. The FAB will only be displayed when the user flings-up at the bottom of the list of items.

Adding or editing a schedule item will open a popup dialog. The dialog will contain Date and Time Picker components and a Spinner component to select the date, time and routine, respectively, for that scheduled session.

Client History Sessions Screen



The Client History Sessions screen is where the user can see the client's previous training sessions. It displays the date and time of the session and the list of exercises and number of sets done for each exercise.

The Client History Sessions screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, ViewPager, and RecyclerView.

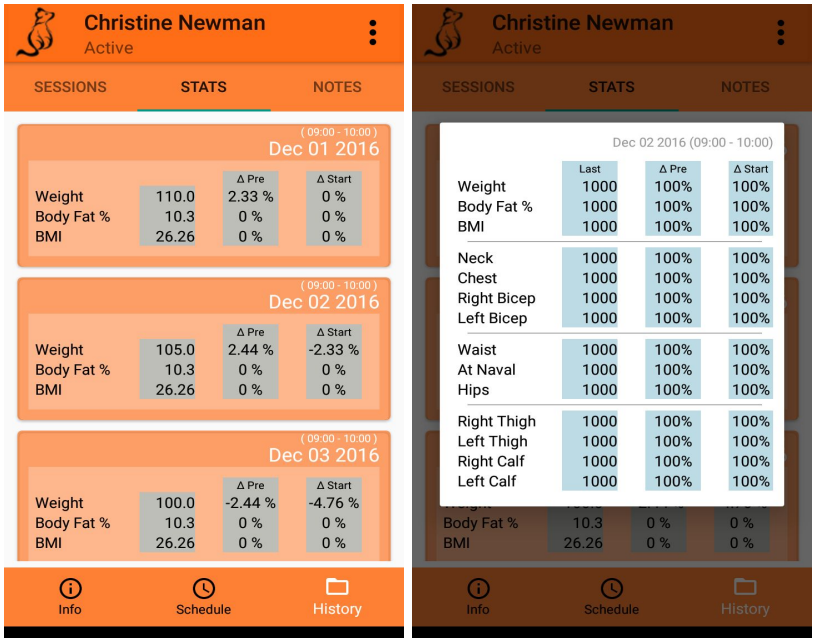
The Bottom Navigation allows the user to switch between the Client Info, Client Schedule and Client History screens.

The ViewPager allows the user to view the Client History Sessions, Stats, and Notes screen

The RecyclerView displays a summary of training sessions taken. Clicking on an item will bring up a dialog displaying more information about the training session.

The dialog displays the date and time of the session, a selected exercise from the exercises done in that session and the recorded details of that exercise (reps, weight, time, distance). A spinner component is used to allow the user to select which exercise to view.

Client History Stats Screen



The Client History Stats screen is where the user can see the client's previous measured statistics. It displays the date and time of the session and weight, body fat % and BMI taken during that session and the percentage from the previous measurements and start measurements.

The Client History Sessions screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, ViewPager, and RecyclerView.

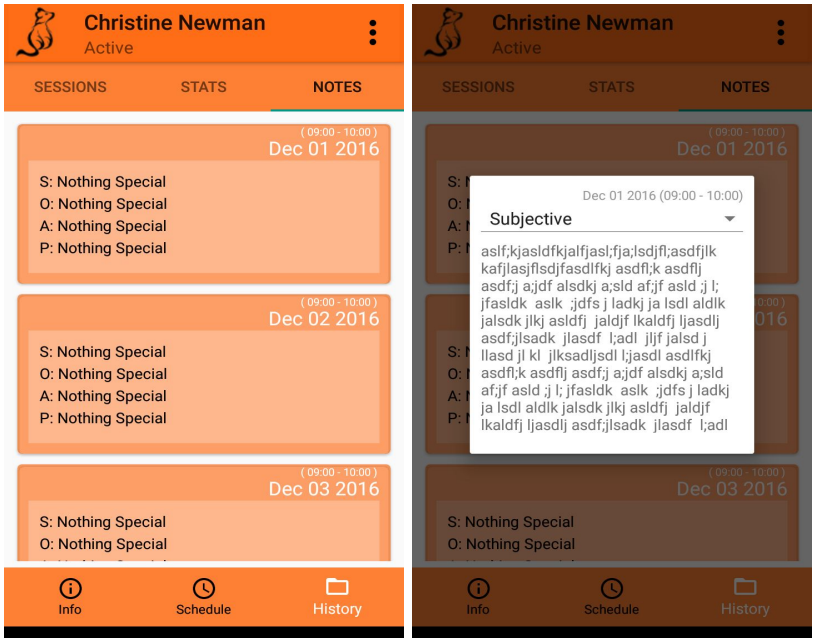
The Bottom Navigation allows the user to switch between the Client Info, Client Schedule and Client History screens.

The ViewPager allows the user to view the Client History Sessions, Stats, and Notes screen

The RecyclerView displays a subset of the measurements taken. Clicking on an item will bring up a dialog displaying more measurements taken during that session.

The dialog displays the date and time of the session, and body measurements that are generally taken.

Client History Notes Screen



The Client History Notes screen is where the user can see the client's previous SOAP notes. It displays the date and time of the session and a brief summary of each note.

The Client History Notes screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, ViewPager, and RecyclerView.

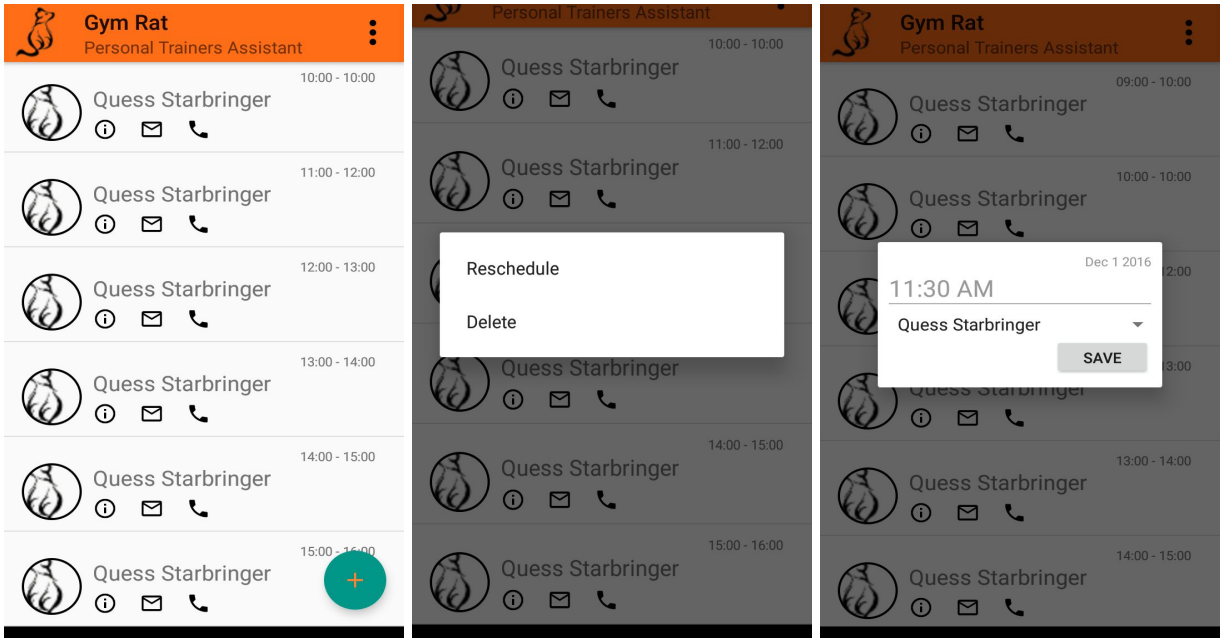
The Bottom Navigation allows the user to switch between the Client Info, Client Schedule and Client History screens.

The ViewPager allows the user to view the Client History Sessions, Stats, and Notes screen

The RecyclerView displays a brief summary of each SOAP note. Clicking on an item will bring up a dialog displaying more measurements taken during that session.

The dialog displays the date and time of the session and a selected note. A spinner component is used to allow the user to select which note to view (Subjective, Objective, Assessment, or Plan).

Sessions Screen



The Sessions screen is where the user can see the list of client sessions he/she has for the day. It displays the date and time of the session, the client name and photo, access to the Client Info screen, and can contact the client via email, sms or phone.

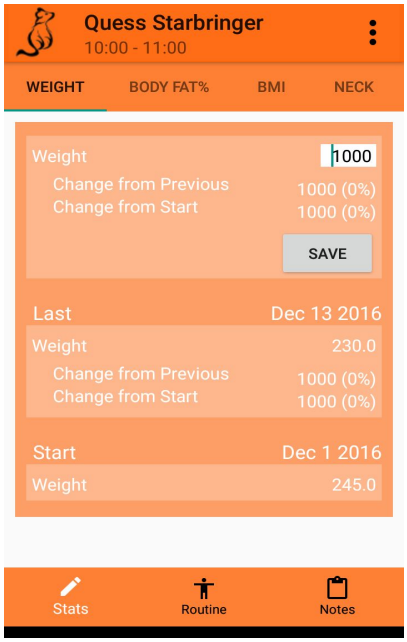
The Sessions screen contains a Toolbar (see Main Navigation for more info), RecyclerView and FAB component.

The RecyclerView displays the list of clients scheduled for training session for the day. Clicking on an item will take the user to the Sessions Detail screen. Long-clicking on an item will open a context menu that will give the user the option to reschedule or delete a session.

The FAB component allows the user to add a new session to the list. The FAB will only be displayed when the user flings-up at the bottom of the list of items.

Adding or editing a session item will open a popup dialog. The dialog will contain Date and Time Picker components and a Spinner component to select the date, time and client, respectively, for that scheduled session.

Sessions Stats Screen



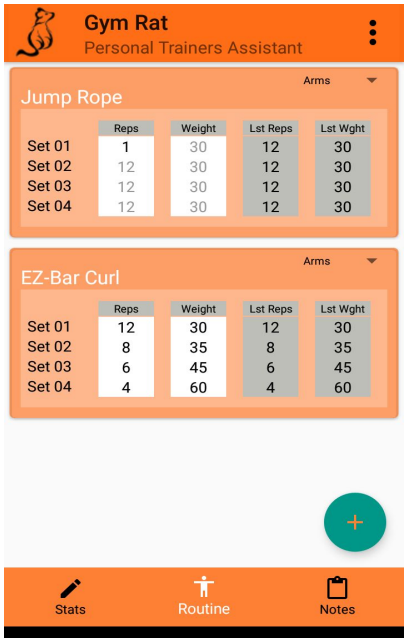
The Sessions Stats screen is where the user can record measurement statistics taken during the session. When taking measurements, the user will also be able to see the first and last measurements and get will get percentage calculations compared to the current measurement.

The Sessions Stats screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, and ViewPager.

The Bottom Navigation allows the user to switch between the Sessions Stats , Sessions Routine and Sessions Note screens.

The ViewPager allows the user to view the following measurement statistics: weight, body fat %, bmi, neck, chest, right bicep, left bicep, waist, navel, hips, right thigh, left thigh, right calf and left calf

Sessions Routine Screen



The Sessions Routine screen is where the user can record details of the exercises performed in the session. The exercise can either be predefined before the start of the session or exercises can be added during the session.

The Sessions Stats screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, ViewPager, and FAB.

The Bottom Navigation allows the user to switch between the Sessions Stats , Sessions Routine and Sessions Note screens.

The FAB component allows the user to add a new exercise to the list. The FAB will only be displayed when the user flings-up at the bottom of the list of items.

Adding an exercise item in the routine will open a popup dialog. The dialog will contain Spinner options for exercise categories, exercises within the selected category and number of sets to be done during the exercise routine (see popup dialog in Edit Routine screen).

Sessions Notes Screen

The Sessions Notes screen is where the user can record notes of the session. The user will also be able to see the last notes taken.

The Sessions Stats screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, and ViewPager.

The Bottom Navigation allows the user to switch between the Sessions Stats , Sessions Routine and Sessions Note screens.

The ViewPager allows the user to view the following notes: Subjective, Objective, Assessment, Plan

Appointment Day Screen

The Appointment Day screen is where the user can see the list of client sessions he/she has for a particular day. It displays the date and time of the session, the client name and photo, access to the Client Info screen, and can contact the client via email, sms or phone.

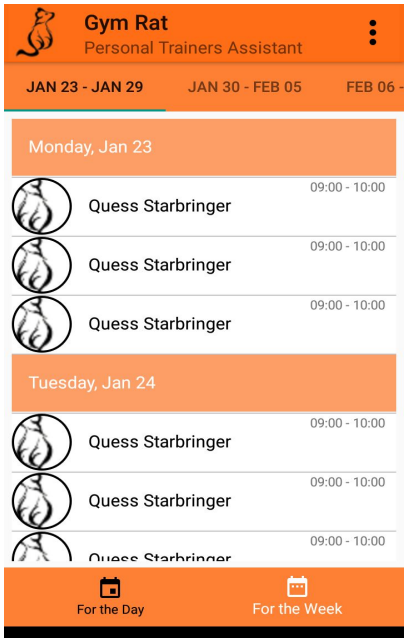
The Appointment Day screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, ViewPager, RecyclerView and FAB component.

The Appointment Day screen contains everything the Sessions screen has with the addition of a ViewPager component (see Sessions screen for details)

The Bottom Navigation allows the user to switch between the Appointment Day and Appointment Week screens.

The ViewPager allows the user to view what appointments he/she has in a given day.

Appointment Week Screen



The Appointment Week screen is where the user can see the list of client sessions he/she has for a particular week. The appointments are separated by the days of the week with each list item containing the client name and photo and time of appointment.

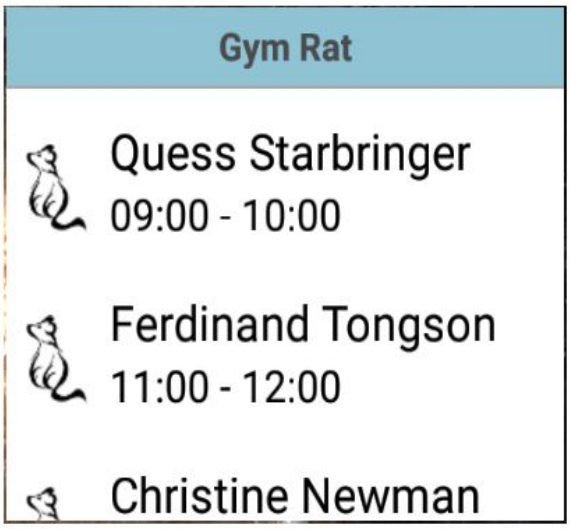
The Appointment Day screen contains a Toolbar (see Main Navigation for more info), Bottom Navigation, ViewPager, and RecyclerView.

The Bottom Navigation allows the user to switch between the Appointment Day and Appointment Week screens.

The ViewPager allows the user to view what appointments he/she has in a given week.

The RecyclerView displays the list of clients scheduled for training session for the week, separated by days. Clicking on an item will take the user to the Client Info screen. Long-clicking on an item will open a context menu that will give the user the option to reschedule or delete a session.

Widget Screen



The Widget screen is where the user can see the list of client sessions he/she has for that particular day. The appointments are ordered by time with each list item containing the client name and photo and time of appointment. Clicking on the title view will open the main application.

The Widget screen contains a ListView.

The ListView displays the list of clients scheduled for a training session for the day.

Key Considerations

How will your app handle data persistence?

Firebase Database, with enabled offline capabilities, will handle the online data storage and retrieval for Gym Rat. SQLite will provide local data storage support.

The app will use Android's Contacts Provider to access and store client contact information and Android's Calendar Provider to handle the scheduling of client appointments.

Finally, Shared Preferences will be used to store the user's setting preferences.

Describe any corner cases in the UX.

Corner Case 01:
In the Exercise Screen, the deletion of an exercise item can potentially affect saved exercise routines in the Routine Screen and in the popup dialog exercise spinner used to add/edit exercise items in a routine. If such a situation arises, the user will be notified and will be given the option to cancel the deletion of the exercise item or to allow Gym Rat to automatically remove the exercise from the routine as well.

Corner Case 02:
In the Exercise Screen, the deletion of an exercise item can potentially affect saved exercise routines in the Routine Screen where the deletion will leave a saved Routine empty of any exercises. If such a situation arises, the user will be notified and will be given the option to cancel the deletion of the exercise item or to allow Gym Rat to automatically remove the routine.

Describe any libraries you'll be using and share your reasoning for including them.

Firebase libraries used for user authentication, real-time database, crash reporting and analytics:

- com.firebaseui:firebase-ui
- com.google.firebase:firebase-auth
- com.google.firebase:firebase-database
- com.google.firebase:firebase-crash
- com.google.firebase:firebase-core

Android libraries to support Material Design, fundamental app components, user interface widgets, backward-compatibility, and utilities:

- com.android.support:appcompat-v7
- com.android.support:design
- com.android.support:recyclerview-v7
- com.android.support:cardview-v7
- com.android.support.constraint:constraint-layout

com.android.support:support-v13
com.android.support:palette-v7

Picasso to handle loading and caching of images:
com.squareup.picasso:picasso

Henning Dodenhof CircleImageView library to create circular imageViews:
de.hdodenhof:circleimageview

ChoiceLibrary.aar to provide MVC / MVP base structure.

Describe how you will implement Google Play Services.

I will be using the following Firebase services: Authentication, Database, Crash Reporting and Analytics.

Firebase Authentication used for user authentication. Gym Rat requires that all users are authenticated before starting to use the app so authentication will be required (checked) at the start of the app. If users are authenticating for the first time, users will have the option to authenticate using their gmail, facebook, twitter or email account. Once authenticated, a unique id will be generated and stored in Firebase Database. This user id will be used to record and access records generated by the user.

Firebase Database used for real-time database. Gym Rat is a Personal Trainer Assistant app that helps personal trainers record and store client information, client training sessions, and client appointments. Firebase Database will function as the online data storage and retrieval of these records. If a personal trainer is using more than one device, Firebase Database will also automatically update those other devices in real-time, if those devices are online.

Firebase Crash Reporting for app crash reporting. Firebase Crash Reporting will track app crashes and exceptions within Gym Rat. These reports will be used to track and resolve bugs.

Firebase Analytics for app analytics. Firebase Analytics will track how users are interacting with Gym Rat. These analytics will be used to help improve user experience and engagement.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and decompose them into tangible technical tasks that you can complete incrementally until you have a finished app.

Task 1: Project Setup

Add ChoiceLibrary.arr to project

- Add “libs” directory to “app” directory
- Copy ChoiceLibrary.aar to “libs” directory
- In app build.gradle between dependencies{ }, check / add the following:
    - compile fileTree(dir: ‘libs’, include: [‘*.jar’])
    - compile(name: ‘ChoiceLibrary’, ext: ‘aar’)
- In project level build.gradle (in this case GymRatAssistant) between allprojects{ repositories { } }, check / add the following:
    - flatDir{ dirs ‘libs’ }
- In project level settings.gradle, check / add the following:
    - Include ‘:app’, ‘:ChoiceLibrary’

Add libraries supporting libraries

- In app build.gradle between dependencies{ }, check / add the following
    - compile ‘com.firebaseui:firebase-ui:1.0.x’
    - compile ‘com.google.firebase:firebase-auth:10.0.x’
    - compile ‘com.google.firebase:firebase-database:10.0.x’
    - compile ‘com.google.firebase:firebase-crash:10.0.x’
    - compile ‘com.google.firebase:firebase-core:10.0.x’
    - compile ‘com.android.support:appcompat-v7:25.0.x’
    - compile ‘com.android.support:design:25.0.x’
    - compile ‘com.android.support:recyclerview-v7:25.0.x’
    - compile ‘com.android.support:cardview-v7:25.0.x’
    - compile ‘com.android.support.constraint:constraint-layout:1.0.x’
    - compile ‘com.android.support:support-v13:25.0.x’
    - compile ‘com.android.support:palette-v7:25.0.x’
    - compile ‘com.squareup.picasso:picasso:2.5.x’
    - compile ‘de.hdodenhof:circleimageview:2.1.x’
- In app build.gradle at the root, check / add the following to enable firebase
    - Apply plugin: ‘com.google.gms.google-services’

Prevent lint error from crashing app

- In app build.gradle between android{ }, check / add
    - lintOptions{ abortOnError false }

Task 2: Basic UI Scaffolding

- Create simple layout
    - One textView component
    - Toolbar component
    - Drawer component
- Create activities with layout
    - Clients Activity
    - Appointments Activity
    - Sessions Activity
    - Exercise Activity
- Create Toolbar with following components
    - Navigation icon
    - Title and subtitle textViews
    - Options icon
- Create Drawer with menu options to the simple activities
    - Menu item clicks will navigate user to selected activity

Task 3: Appointment UI Scaffolding

- Create UI for Appointment Activity as described in Appointment Day and Appointment Week screen
- Create data stubs to simulate data used by each screen

Task 4: Sessions UI Scaffolding

- Create UI for Sessions Activity as described in Sessions screen
- Create data stubs to simulate data used by the screen

Task 5: Clients UI Scaffolding

- Create UI for Clients Activity as described in Clients screen
- Create data stubs to simulate data used by the screen

Task 6: Exercises UI Scaffolding

- Create UI for Exercises Activity as described in Exercise and Routine screen
- Create data stubs to simulate data used by each screen

Task 7: Permissions

- Add the following permission in app AndroidManifest.xml between <manifest></manifest> (NOTE: do NOT put permissions between <application></application>)
    - <uses-permission android:name=”android.permission.INTERNET” />
    - <uses-permission android:name=”android.permission.ACCESS_NETWORK_STATE” />
    - <uses-permission android:name=”android.permission.READ_CONTACTS” />
    - <uses-permission android:name=”android.permission.READ_CALENDAR” />

    - <uses-permission android:name="android.permission.WRITE_CALENDAR" />
- Create Permissions helper class to handle user permission requests

Task 8: Firebase Setup

- Create Firebase project
    - Set up Database, Authentication
    - Add Crash Reporting and Analytics to Activities
- Add Firebase authentication at launch of Gym Rat to Sign in
- Sign out using toolbar

Task 9: Exercise Default Data

- Create default exercise category and exercise data as a flat file
- Create SQLite tables to hold default exercise data
- At first-launch,
    - Create SQLite database and load default exercise data
    - Create Firebase tables and load default exercise data
- Load data to UI
- Test add, edit and delete functionality
    - Check both SQLite and Firebase are being properly updated

Task 10: Routine Default Data

- Create default exercise routine data as a flat file
- Create SQLite tables to hold default exercise routine data
- At first-launch,
    - Create SQLite table and load default exercise routine data
    - Create Firebase tables and load default exercise routine data
- Load data to UI
- Test add, edit and delete functionality
    - Check both SQLite and Firebase are being properly updated

Task 11: Clients and Contact Data

- Read user's contact data and display in dialog fragment
- Create SQLite tables to hold necessary contact/client data
- At first-launch,
    - Create SQLite table for client info data
- Add contact to client list
    - Add contact info to SQLite database
    - Add contact info to Firebase database
- Test add functionality
    - Check both SQLite and Firebase are being properly updated

Task 12: Clients UI Scaffolding, part 2

- Create UI for Clients Activity as described in the following screens
    - Client Info Screen

    - Client Schedule Screen
    - Client History Sessions Screen
    - Client History Stats Screen
    - Client History Notes Screen
- Create data stubs to simulate data used by the screens
- At first-launch,
    - Create SQLite tables
    - Create Firebase tables

Task 13: Client Info Data

- Clicking on a client in the Client Screen will take the user to the Client Info Screen
- Populate screen with client info data taken from database
- Clicking on email icon will enable user to send an email to the client
    - Send test email to confirm functionality
- Clicking on phone icon will enable user to send an sms message or call the client
    - Test sms and phone call
- Add client's training goals
    - Check both SQLite and Firebase are being properly updated

Task 14: Client Schedule Data

- Add and delete a training session for the client
    - Check both SQLite and Firebase are being properly updated
- Make sure the client can NOT book a session in conflict with the client's schedule
- Add client session to the user's google calendar
    - Check for scheduling conflicts before booking client

Task 15: Client History Data

- Load temporary client history data to SQLite database
- Use data from database to populate screens
    - Check that detailed history information displayed by dialogs are accurate

Task 16: Appointment Day Data

- Add and delete a training session for a client
    - Check both SQLite and Firebase are being properly updated
- Make sure the client can NOT book a session in conflict with the client's schedule
- Add client session to the user's google calendar
    - Check for scheduling conflicts before booking client

Task 17: Appointment Week Data

- Using the appointments created in Task 16
    - Check that appointments are properly displayed

Task 18: Appointment Widget

- Create appointment widget

    - Check that appointments for that day are properly displayed
    - Check that the widget is updated with the addition or subtraction of an appointment

Task 19: Sessions Data

- Add and delete a training session for a client
    - Check both SQLite and Firebase are being properly updated
- Make sure the client can NOT book a session in conflict with the client's schedule
- Add client session to the user's google calendar
    - Check for scheduling conflicts before booking client
- Make sure the user can NOT book a session at a time that has already passed
- Test email and phone icon functionality
- Clicking on info icon will take user to the client's info page
    - Back button will return user to Sessions screen

Task 20: Sessions UI Scaffolding part 2

- Create UI for Sessions Activity as described in the following screens
    - Sessions Stats Screen
    - Sessions Routine Screen
    - Sessions Notes Screen
- Create data stubs to simulate data used by the screens
- At first-launch,
    - Create SQLite tables
    - Create Firebase tables

Task 21: Sessions Stats, Notes and Routine Data

- Add session stats for a client
    - Check both SQLite and Firebase are being properly updated
- Add session notes for a client
    - Check both SQLite and Firebase are being properly updated
- Add session routine for a client
    - Check both SQLite and Firebase are being properly updated

Task 22: Quick Help and Help

- Add Quick Help dialog information for each screen / fragment
- Clicking on Help in the Drawer layout will send the user to a website with more info

Task 23: Settings

- Create popup setting dialog
    - Allow user to hide Sessions Stats and/or Sessions Notes screen
    - Allow user to enable or disable sessions notification alert
- Check both SQLite and Firebase are being properly updated

Task 24: Debugging and Testing

- Check UI flow
- Debug appointment setting and client scheduling, check for conflicts
- Double check records are being properly displayed and calculated
- Using two devices, test that the second device is being dynamically updated