Lifting Log Test Specification

Joe Melito & Neil Kalanish

1.0 Introduction

Our application, Lifting Log, heavily focuses on accessing and modifying our back end database. We want our users to have the ability to create new records, and view / modify previous records. Our test cases will be dependent on successfully accessing our database, presenting user requested data appropriately, and allowing users to seamlessly navigate through our UI.

1.1 Goals and objectives

We want our users to be presented with the expected data that they are requesting from our database, have access to accurate 1RM & Wilks calculations, and have the ability to successfully create, modify, & view account information.

1.2 Statement of scope

The Following Test Cases Will Be Within Scope:

- Account Creation & Deletion
- Login Credentials
- Navigation Between All Views Within Our UI
- Accurately Displaying Requested Data (Viewing Previous Workouts)
- Accurate Submission of User Data to Our Database.
- Accurate Calculations (1RM & Wilks Calculators)
- Users Ability To Enter In The Expected Data (User Enters in String When Integer is Expected)

The Following Test Cases Will Not Be Within Scope:

- Unit Testing of Data Transactions & Calculations
- User Gets Phone Call or Text During Usage (I.E. User Leaves Our Application Mid Flow)
- User's Phone Battery Dies
- User's Phone Crashes Due to OS Related Issues
- User Upgrades or Downgrades OS on Device
- User is using an Unreleased or Beta Version of OS
- User Loses Internet Connectivity

2.0 Test Plan

As mentioned previously, due to the constraints of developing an application within a 16 week period we will not be including unit testing within our application scope. Therefore, our test cases will be done manually from a viewpoint of a QA tester. We will break down our test cases in section 3.0.

2.1 Software to be tested

Lifting Log is our application which we are developing using Apple's programming language, Swift, and allowing it to have access to Google's Database Engine, Firebase. We are primarily focusing on testing our application and how it interfaces with the mentioned technologies.

2.3 Testing tools and environment

A description of test environment, including tools, simulators, specialized hardware, test files, and other resources is presented here.

We will conduct testing on our local machines by running our solution through Apple's IDE Xcode. Xcode, gives developers the ability to simulate any supported Apple Device; allowing us to test our application on iPhone, iPad, Mac, & Apple Watch. We also are going through the process of registering our personal devices as developer devices so that we can test our application on real hardware. As for the files we will be using to test. These include:

Services:

- Calculation Service: Handles our 1RM and Wilks Calculations.
- Alert Service: Handles user prompted alerts.
- Authentication Service: Handles authenticating our users accounts. (User login and account creation.)
- Database Service: Handles access and modification functionally between our application, the user, and our database.

Extensions:

- UI View Controller Extension: Creates additional functionality for our UI.

Views:

- One RM Calculation View: Handles UI for our 1RM Calculator.
- Wilks Calculation View: Handles UI for our Wilks Calculator.
- View Controller: Handles UI for our user login.
- Create Account Controller: Handles UI for user account creation.
- Create Workout Controller: Handles UI for creating a workout so the user may come back to it later.

- New Workout View: Handles UI for creating a new workout.
- Past Workout Controller: Handles UI for accessing previous submitted workouts.
- Settings Controller: Handles UI for user based application settings.
- Welcome Dashboard Controller: Handles UI for our main dashboard which allows the user to navigate and access all corresponding functionality within our application.

2.4 Test schedule

View	Task	Members	Estimated Time	Deadline	Status
User Login	No account	Backend	2-3 Hours	10/31/22	Complete
User Login	Wrong username or password	Backend	2-3 Hours	10/31/22	Complete
User Login	All inputs are responsive	Front end	1 Hour	10/31/22	Complete
Create Account	Account creation successful	Backend	2-3 Hours	10/31/22	Complete
Create Account	All inputs are responsive	Front end	1 Hour	10/31/22	Complete
Create Account	Account creation cancellation	Backend	2-3 Hours	10/31/22	Complete
Main Dash	All inputs are responsive	Front end	1 Hour	10/31/22	Complete
1RM Calculator	All inputs are responsive	Front end	1 Hour	11/18/22	In Process
1RM Calculator	Output calculated correctly	Backend	2-3 Hours	11/18/22	In Process
Wilks Calculator	All inputs are responsive	Front end	1 Hour	11/18/22	In Process
Wilks Calculator	Output calculated correctly	Backend	2-3 Hours	11/18/22	In Process
Create Workout	All inputs are responsive	Front end	1 Hour	10/31/22	Complete
Create Workout	Data properly saved for future use	Backend	2-3 Hours	11/25/22	Not Started
New Workout	All inputs are responsive	Front end	1 Hour	11/25/22	Not Started
New Workout	Upon submission data is stored in database.	Backend	2-3 Hours	11/25/22	Not Started
Past Workout	All inputs are responsive	Front end	1 Hour	11/25/22	Not Started
Past Workout	Data retrieved is accurate	Backend	2-3 Hours	11/25/22	Not Started
Settings	All inputs are responsive	Front end	1 Hour	10/31/22	Complete
Settings	Logout feature works as expected	Backend	2-3 Hours	10/31/22	Complete
Settings	Account deletion works as expected	Backend	2-3 Hours	10/31/22	Complete

3.0 Test Cases

ID	User Login: No Account
Test Input	Username: TEST1234@Test.com Password: TEST123!
Expected Output	User will be prompted with a dialog box stating that their account cannot be found.
Description	User enters in 'account information' without registering an account. User should be notified that signup is required.

ID	User Login: Wrong Username & Password
Test Input	Username: dev Password: TEST123!
Expected Output	User will be prompted with a dialog box stating that their login credentials are incorrect.
Description	User enters wrong 'account information' User should be notified that their password and / or username is incorrect. User must be asked to try again.

ID	User Login: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	Create Account: Account Creation Successful
Test Input	Tester may enter any values for account creation.
Expected Output	Account should be created and allow the user to login.
Description	Users must be able to create an account / sign up for our application.

ID	Create Account: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	Create Account: Account Creation Cancellation
Test Input	Tester may enter any input values.
Expected Output	When a user cancels account creation; no account is created and they are redirected to the login screen.
Description	Users must be able to cancel account creation if they so wish.

ID	Main Dash: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	1RM Calculator: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	1RM Calculator: Output Calculated Correctly
Test Input	Tester may enter any input values.
Expected Output	An integer output value.
Description	Users must be able to enter in the corresponding data for 1RM and receive and accurate output value.

ID	Wilks Calculator: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	Wilks Calculator: Output Calculated Correctly
Test Input	Tester may enter any input values.
Expected Output	An integer output value.
Description	Users must be able to enter in the corresponding data for Wilks and receive and accurate output value.

ID	Create Workout: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	Create Workout: Data Properly Saved For Future Use
Test Input	Tester may enter any input values.
Expected Output	Inputted values will be saved for future use
Description	Users must be able to create a workout, input data, save data, and come back to it at a later time for modification or submission.

ID	New Workout: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	New Workout: Data Submission To Database
Test Input	Tester may enter any input values.
Expected Output	All inputted data must be shown within our database
Description	Users must be able to input and submit workout data to our database.

ID	Past Workout: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	Past Workout: Data Retrieved is Accurate
Test Input	Tester may select a previous workout
Expected Output	Data retrieved must match user’s previous workout data submissions.
Description	Users must be able to view previous submitted workout data.

ID	Settings: All Inputs Are Responsive
Test Input	Tester may enter any input values.
Expected Output	All input fields should be responsive.
Description	Users must be able to interact with this part of the application.

ID	Settings: Logout Feature
Test Input	Tester may select the ‘Logout’ button.
Expected Output	User will be logged out of application
Description	Users must be able to securely log out of our application.

ID	Settings: Account Deletion Feature
Test Input	Tester may select the ‘Delete Account’ button.
Expected Output	User will be prompted with a dialog box warning them; if they hit ‘cancel’ they remain logged in; if they hit ‘ok’ their account information will be removed from the database
Description	Users must be able to delete their accounts when they no longer wish to use lifting log.