Dithi Saxena
Catherine Penquite
Abhishek Gunasekar
Christopher Yu
Vivek Nair
Team 6

Sprint 3 Testing

21st April 2021

Werk It

TABLE OF CONTENTS

Manual Tests

User Story #1

As a user, I would like to be able to edit previously saved workouts, workout plans, and workout types.

1. **Manual Test #1**
    a. Identification and Classification
        Test Case 01
        System: WorkoutEditor.js   Phase: 2
        Check if user can edit the workout type
        Severity: 1
    b. Instructions
        i. The user is on the workout editor screen and opens the workout type dropdown menu and selects a workout type they want to edit.
    c. Expected Result
        i. The user should be given the option to edit that workout type selected on the mobile screen and an editable text box should appear with the name of the workout type already filled.
2. **Manual Test #2**
    a. Identification and Classification
        Test Case 02
        System: WorkoutEditor.js   Phase: 2
        Check if the edit workout type successfully saves the new data in the database.
        Severity: 1
    b. Instructions
        i. The user has selected a workout type to edit and an editable text box appears with the name of the workout type already filled, and the user edits the workout and re-submits it.
    c. Expected Result
        i. The user should be able to see the workout type is successfully updated if the edit button is chosen again, and the database has successfully stored the new information for the workout type.
3. **Manual Test #3**
    a. Identification and Classification
        Test Case 03
        System: WorkoutEditor.js/WorkoutPlanEditor.js   Phase: 2

Check if the user can edit a workout/workout plan.

Severity: 1

b. Instructions

   i. The user is on the workouts/workout plans screen and taps and of the workouts/workout plans displayed.

c. Expected Result

   i. The user should be given the option to edit the workout/workout plan selected on the mobile screen and an editable text box should appear with existing information already populated in any information fields.

4. **Manual Test #4**

   a. Identification and Classification

   Test Case 04

   System: WorkoutEditor.js/WorkoutPlanEditor.js Phase: 2

   Check if the edit workouts/workout plans successfully saves the new data in the database.

   Severity: 1

   b. Instructions

      i. The user has selected a workout/workout plan to edit and the user edits the workout/workout plan and re-submits it.

   c. Expected Result

      i. The user should be able to see the workout/workout plan is successfully updated if the edit button is chosen again, and the database has successfully stored the new information for the workout/workout plan.

User Story #2

As a user, I would like to be able to delete previously created workouts, workout plans, workout types, and exercises.

1. **Manual Test #1**
    a. Identification and Classification
    Test Case 01
    System: WorkoutEditor.js/WorkoutPlanEditor.js Phase: 2
    Check if a user is able to delete a workout/workout plan.
    Severity: 1
    b. Instructions
        i. The user is on the workouts/workout plans screen and taps any of the workouts/workout plans displayed.
    c. Expected Result
        i. The user should be given the option to delete the workout/workout plan selected on the mobile screen.
2. **Manual Test #2**
    a. Identification and Classification
    Test Case 02
    System: WorkoutEditor.js/WorkoutPlanEditor.js Phase: 2
    Check if the delete workouts/workout plans successfully deletes existing data in the database.
    Severity: 1
    b. Instructions
        i. The user has selected a workout/workout plan to delete and the user deletes the workout/workout plan.
    c. Expected Result
        i. The user should not be able to see the workout/workout plan deleted on the mobile screen if it is successfully deleted, and the database has successfully deleted the information for the workout/workout plan the user selected to delete.
3. **Manual Test #3**
    a. Identification and Classification
    Test Case 03
    System: WorkoutEditor.js Phase: 2
    Check if the user can delete a workout type or exercise.
    Severity: 1

b. Instructions
   i. The user is on the workout editor screen and opens the workout type dropdown menu or opens the exercise editor menu and selects what workout type/exercise they would like to delete.

c. Expected Result
   i. The user should be given the option to delete that workout type/exercise selected on the mobile screen.

4. **Manual Test #4**

   a. Identification and Classification

      Test Case 04

      System: WorkoutEditor.js Phase: 2

      Check if the delete workout type/exercise successfully deletes existing data in the database.

      Severity: 1

   b. Instructions
      i. The user is on the workout editor screen and opens the workout type dropdown menu or opens the exercise editor menu and selects what workout type/exercise they would like to delete.

   c. Expected Result
      i. The user should not be able to see the workout type/exercise deleted on the mobile screen if it is successfully deleted, and the database has successfully deleted the information for the workout type/exercise the user selected to delete.

User Story #3

As a user, I would like to be able to modify the user information on the profile page.

1. **Manual Test #1**
    a. Identification and Classification
       Test Case 05
       System: profile.html Phase: 2
       Severity: 2
       Check if the profile page displays user information.
    b. Instructions
        i. The user is logged in on the web app and visits the profile page from the dashboard page.
    c. Expected Result
        i. Web client should redirect user to the profile page which displays the name, username, email, and location of the current user in read only text fields.
2. **Manual Test #2**
    a. Identification and Classification
       Test Case 06
       System: profile.html Phase: 2
       Check if the user can edit information on the profile page.
       Severity: 2
    b. Instructions
        i. The user is logged in on the web app and visits the profile page.
    c. Expected Result
        i. Web clients should show a pen icon next to each read only text field on the profile page, which when clicked makes the read only text field editable.
3. **Manual Test #3**
    a. Identification and Classification
       Test Case 07
       System: profile.html Phase: 2
       Check if the edited profile information is correctly displayed on the profile page.
       Severity: 2
    b. Instructions
        i. The user is logged in on the web app, visits the profile page, edits some profile information with the pen icon, and saves their preferences.

   c. Expected Result
      i. Web client should update text fields to display newly edited information on the profile screen.

4. **Manual Test #4**
   a. Identification and Classification

      Test Case 08

      System: profile.html   Phase: 2

      Check if edited profile information is saved to the database.

      Severity: 2
   b. Instructions
      i. The user is logged in on the web app, visits the profile page, edits some profile information with the pen icon, and saves their preferences.
   c. Expected Result
      i. The backend route should successfully update the user’s profile information in the database.

User Story #4

As a user, I would like to see a motivational quote whenever I access the app.

1. **Manual Test #1**
    a. Identification and Classification
       Test Case 09
       System: MotivationalQuote.js, quote.html Phase: 3
       Check if a motivational quote is displayed on both clients when logging into the application.
       Severity: 2
    b. Instructions
        i. User opens the web or mobile client and logs in.
    c. Expected Result
        i. Both clients should display a randomized motivational quote when the user is being logged into the application.
2. **Manual Test #2**
    a. Identification and Classification
       Test Case 10
       System: App.js , quote.html Phase: 3
       Check if user is redirected to the dashboard after 5 seconds of the motivational quote showing
       Severity: 2
    b. Instructions
        i. User opens the web or mobile client and logs in. Both clients should display a randomized motivational quote when the user is being logged into the application.
    c. Expected Result
        i. Both clients should redirect the user to the dashboard after 5 seconds of the motivational quote showing.
3. **Manual Test #3**
    a. Identification and Classification
       Test Case 11
       System: MotivationalQuote.js, quote.html Phase: 3
       Check that the motivational quote appears when the mobile app is loading data in the background
       Severity: 2
    b. Instructions

        i. The user is on the mobile app and logged in.

    c. Expected Result

        i. Mobile client should display motivational quote whenever the app is loading data in the background (transitioning between pages, logging in, loading data, etc.).

4. **Manual Test #4**
    a. Identification and Classification

       Test Case 12

       System: MotivationalQuote.js, quote.html Phase: 3

       Check if a motivational quote is displayed in a randomized order and is not the same one shown always.

       Severity: 2

    b. Instructions
        i. User opens the web or mobile client and logs in and logs out multiple times.
    c. Expected Result
        i. Every time the user logs in and logs out on the web or mobile client, a random motivational quote should be displayed.

User Story #5

As a user, I would like to be able to easily comprehend the visualizations generated based on my workout statistics.

1. **Manual Test #1**
    a. Identification and Classification

       Test Case 13

       System: dashboard.html Phase: 2

       Check if the pie chart is empty if the user has not completed any workouts for the week.

       Severity: 1
    b. Instructions
        i. The user is on the web client on the dashboard page and has not completed any workouts for the week.
    c. Expected Result
        i. The web client pie chart should be empty.
2. **Manual Test #2**
    a. Identification and Classification

       Test Case 14

       System: dashboard.html Phase: 2

       Check if the pie chart shows accurate percentages.

       Severity: 1
    b. Instructions
        i. The user is on the web client on the dashboard page and has completed at least one workout in the week.
    c. Expected Result
        i. The web client pie chart should show the percentage of workout time for each day of the week.
3. **Manual Test #3**
    a. Identification and Classification

       Test Case 15

       System: dashboard.html Phase: 2

       Check if the column chart shows accurate data.

       Severity: 1
    b. Instructions
        i. The user is on the web client on the dashboard page.
    c. Expected Result

i. The web client column chart should show the number of workout types completed over the course of the year.

4. **Manual Test #4**
    a. Identification and Classification
    Test Case 16
    System: dashboard.html Phase: 2
    Check if the geo chart shows accurate data.
    Severity: 1
    b. Instructions
        i. The user is on the web client on the dashboard page.
    c. Expected Result
        i. The web client geo chart should show the locations of the user’s friends that use the app.

User Story #6

As a user, I would like to be able to connect with my friends.

1. **Manual Test #1**
    a. Identification and Classification

       Test Case 17

       System: Friends.js Phase: 2

       Check if "Friends" option is available on the drawer on the mobile app

       Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app and taps the drawer icon on the top right of the screen.
    c. Expected Result
        i. The mobile client should show an option for the "Friends" screen on the drawer.
2. **Manual Test #2**
    a. Identification and Classification

       Test Case 18

       System: Friends.js Phase: 2

       Check if user can add friends when on the "Friends" mobile screen

       Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app, on the friends screen, enters a friends' username to add, and taps "add a friend."
    c. Expected Result
        i. The mobile client should call the backend route to successfully add the friend, and the new friend should be added to the list on the "Friends" screen.
3. **Manual Test #3**
    a. Identification and Classification

       Test Case 19

       System: Friends.js Phase: 2

       Check if the current user's friends have the current user's name on their friends list.

       Severity: 1
    b. Instructions
        i. The user is on a mobile client on the "Friends" screen and has connected with a friend.

   c. Expected Result
      i. When the friend logs in and goes to the “Friends” screen on the mobile app, then the current user’s name should be on the friends list.

4. **Manual Test #4**
   a. Identification and Classification
      Test Case 20
      System: Friends.js Phase: 2
      Check if the user cannot add an existing friend or non-existent user in the database.
      Severity: 1
   b. Instructions
      i. The user is on the mobile client on the friends screen and tries to add an existing friend or a user that does not exist in the database.
   c. Expected Result
      i. The mobile client will give an error message that “You are already friends with this user” or “This user does not exist,” respectively.

User Story #7

As a user, I would like to be able to view the mobile application in dark mode.

1. **Manual Test #1**
    a. Identification and Classification
    Test Case 21
    System: Settings.js Phase: 3
    Check if there is a dark mode toggle on the mobile client.
    Severity: 2
    b. Instructions
        i. The user is on the mobile app and clicks the drawer and goes to the settings screen.
    c. Expected Result
        i. The mobile client should display a toggle option for enabling and disabling dark mode.
2. **Manual Test #2**
    a. Identification and Classification
    Test Case 22
    System: Settings.js Phase: 3
    Check if the dark mode toggle is rendered when dark mode is on.
    Severity: 2
    b. Instructions
        i. The user is on the mobile app on the settings screen and toggles the dark mode option so dark mode is on.
    c. Expected Result
        i. Mobile client should be rendered in dark mode colors for all pages.
3. **Manual Test #3**
    a. Identification and Classification
    Test Case 23
    System: Settings.js Phase: 3
    Check if the dark mode toggle is rendered when dark mode is off.
    Severity: 2
    b. Instructions
        i. The user is on the mobile app on the settings screen and toggles the dark mode option so dark mode is off.
    c. Expected Result
        i. Mobile client should be rendered in light mode colors for all pages.
4. **Manual Test #4**

a. Identification and Classification

Test Case 24

System: Settings.js Phase: 3

Check if dark mode preference is saved upon log in and log out.

Severity: 1

b. Instructions

   i. The user is on the mobile app and has saved dark mode to on and logs out.

c. Expected Result

   i. The mobile client should save the dark mode preference accordingly and when the user is logged back in all pages should be in dark mode.

User Story #8

As a user, I would like to be able to start a workout session.

1. **Manual Test #1**
    a. Identification and Classification
    Test Case 25
    System: Dashboard.js Phase: 2
    Check if there is a option to start the workout of the day
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app on the dashboard screen and has an active workout plan.
    c. Expected Result
        i. The mobile client should display an option to start the workout for the day if there is one.
2. **Manual Test #2**
    a. Identification and Classification
    Test Case 26
    System: Dashboard.js/WorkoutTracker.js Phase: 2
    Check if the user is navigated to the workout tracker screen when a workout is started.
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app and starts a workout.
    c. Expected Result
        i. The mobile client should redirect the user to the workout tracker screen where the exercises for that workout are displayed.
3. **Manual Test #3**
    a. Identification and Classification
    Test Case 27
    System: WorkoutTracker.js Phase: 2
    Check if the user can check off the corresponding exercise when they finish an exercise.
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app and is on the workout tracker screen and has finished an exercise.
    c. Expected Result

        i. The mobile client should allow the user to check off the corresponding exercise when they have finished an exercise.

4. **Manual Test #4**
    a. Identification and Classification

       Test Case 28

       System: WorkoutTracker.js Phase: 2

       Check if time elapsed and completed status is sent to the database upon completion of a workout.

       Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app and has checked off all the exercises in the workout and has finished the workout.
    c. Expected Result
        i. The mobile client should send the time elapsed and a completed status to the database.

User Story #9

As a user, I would like to be reminded when I have an upcoming workout scheduled.

1. **Manual Test #1**
    a. Identification and Classification
       Test Case 29
       System: Dashboard.js Phase: 2
       Check if the user can see a list of upcoming workouts on the mobile dashboard screen
       Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app, redirected to the dashboard screen, and has upcoming workouts.
    c. Expected Result
        i. The mobile client should display a list of upcoming workouts on the dashboard page.
2. **Manual Test #2**
    a. Identification and Classification
       Test Case 30
       System: Dashboard.js Phase: 2
       Check if the user list of upcoming workouts is ordered by day.
       Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app and redirected to the dashboard screen.
    c. Expected Result
        ii. The mobile client should display a list of upcoming workouts on the dashboard page that are ordered by day and where today's workout is on the top of the list.
3. **Manual Test #3**
    a. Identification and Classification
       Test Case 31
       System: Dashboard.js Phase: 2
       Check if the list of upcoming workouts changes if the active workout plan changes.
       Severity: 1
    b. Instructions

        i. The user is logged in on the mobile app, redirected to the dashboard screen, and changes the active workout plan with the drop down selector.
    c. Expected Result
        i. The mobile client should display a list of upcoming workouts on the dashboard page for the new active workout plan selected.

4. **Manual Test #4**
    a. Identification and Classification
    Test Case 32
    System: Dashboard.js Phase: 2
    Check if “No Upcoming Workouts” is displayed if there are no upcoming workouts.
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app, redirected to the dashboard screen, and has no upcoming workouts.
    c. Expected Result
        i. The mobile client should display “No Upcoming Workouts” where the location of upcoming workouts usually is.

User Story #10

As a user, I would like to see a line graph comparing my workout time with my friends.

1. **Manual Test #1**
    a. Identification and Classification

        Test Case 33

        System: profile.html Phase: 2

        Check if there is no line graph data if the user has not completed any workouts for the week.

        Severity: 1

    b. Instructions
        i. The user is logged in on the web client, on the dashboard screen, and has not completed any workouts for the week.

    c. Expected Result
        i. The web client line graph should not display any data.
2. **Manual Test #2**
    d. Identification and Classification

        Test Case 34

        System: profile.html Phase: 2

        Check if there is no line graph data if the user's friends have not completed any workouts for the week.

        Severity: 1

    e. Instructions
        i. The user is logged in on the web client, on the dashboard screen, and the user's friends have not completed any workouts for the week.

    f. Expected Result
        i. The web client line graph should not display any data.
3. **Manual Test #3**
    a. Identification and Classification

        Test Case 35

        System: profile.html Phase: 2

        Check if the line graph should be the correct data when the user and their friends have completed at least one workout in the week.

        Severity: 1

    b. Instructions
        i. The user is logged in on the web client, on the dashboard screen, and the user and the user's friends have completed at least one workout in the week.

   c. Expected Result
      i. The web client line graph should have one line graph for the user and one line graph for each of the friends who completed at least one workout.

4. **Manual Test #4**
   a. Identification and Classification
      Test Case 36
      System: profile.html Phase: 2
      Check if the line graph is updated when the user completes a workout.
      Severity: 1
   b. Instructions
      i. The user is logged in on the web client and has completed a workout.
   c. Expected Result
      i. The web client line graph for the user should increase the workout time accordingly for the workout that was just completed.

User Story #11

As a user, I would like to be able to send my workout plan to my friends.

1. **Manual Test #1**
    a. Identification and Classification
    Test Case 37
    System: Friends.js Phase: 3
    Check if the "send a workout plan" button shows when a friend's name is tapped.
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app, has friends, and taps the friends name.
    c. Expected Result
        i. The mobile client should show a popup screen that has a button, "send a workout plan."
2. **Manual Test #2**
    a. IIdentification and Classification
    Test Case 38
    System: Friends.js Phase: 3
    Check if there is a new pop up screen when the user chooses to "send a workout plan" to a friend.
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app, has friends, taps the friends name, and taps "send a workout plan."
    c. Expected Result
        i. The mobile client should show a new popup screen that has a dropdown picker, a "send" button, and a "cancel" button.
3. **Manual Test #3**
    a. Identification and Classification
    Test Case 39
    System: Friends.js Phase: 3
    Check if the user can select a workout plan to send from a predefined list.
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app, has friends, taps the friends name, and taps "send a workout plan."

c. Expected Result
    i. The mobile client should show a new popup screen that has a dropdown picker containing a predefined list of workout plans.

4. **Manual Test #4**
    a. Identification and Classification
    Test Case 40
    System: Friends.js Phase: 3
    Check if the friend receives a message
    Severity: 1
    b. Instructions
        i. The user is logged in on the mobile app, has friends, taps the friends name, taps “send a workout plan,” selects a workout plan to send, and clicks “send.”
    c. Expected Result
        i. When the friend logs into the mobile and is redirected to the dashboard screen, the friend should have received a message that contains a workout plan sent from the current user.

User Story #12

As a user, I would like to be able to send fitness challenges to my friends.

1. **Manual Test #1**
    a. Identification and Classification

       Test Case 41

       System: Friends.js Phase: 3

       Check if the “send a fitness challenge” button shows when a friend’s name is tapped.

       Severity: 1

    b. Instructions
        i. The user is logged in on the mobile app, has friends, and taps the friends name.

    c. Expected Result
        i. The mobile client should show a popup screen that has a button, “send a fitness challenge.”

2. **Manual Test #2**
    a. Identification and Classification

       Test Case 42

       System: Friends.js Phase: 3

       Check if there is a new pop up screen when the user chooses to “send a fitness challenge” to a friend.

       Severity: 1

    a. Instructions
        i. The user is logged in on the mobile app, has friends, taps the friends name, and taps “send a fitness challenge.”

    b. Expected Result
        i. The mobile client should show a new popup screen that has a text box, a “send” button, and a “cancel” button.

3. **Manual Test #3**
    a. Identification and Classification

       Test Case 43

       System: Friends.js Phase: 3

       Check if the user can input a number of workouts to complete for the fitness challenge

       Severity: 1

    b. Instructions

        i. The user is logged in on the mobile app, has friends, taps the friends name, and taps “send a fitness challenge.”

    c. Expected Result

        i. The mobile client should show a new popup screen that has a text box which only allows the user to enter up to two digits.

4. **Manual Test #4**

    a. Identification and Classification

       Test Case 44

       System: Friends.js Phase: 3

       Check if the friend receives a message

       Severity: 1

    b. Instructions

        i. The user is logged in on the mobile app, has friends, taps the friends name, taps “send a fitness challenge,” enters the number of workouts to finish first, and clicks “send.”

    c. Expected Result

        i. When the friend logs into the mobile and is redirected to the dashboard screen, the friend should have received a message that contains the fitness challenge sent from the current user.

User Story #13

As a developer, I need to display a loading symbol if the user is disconnected from Wifi.

1. **Manual Test #1**
    a. Identification and Classification

       Test Case 45

       System: loading.js Phase: 3

       Check if loading symbol appears

       Severity: 1

    a. Instructions
        i. The user is connected to a pc with wifi and opens the web client. They then disconnect from wifi after 15-30 seconds.

    b. Expected Result
        i. The loading symbol should appear on the web client.

2. **Manual Test #2**
    a. Identification and Classification

       Test Case 46

       System: loading.js Phase: 2

       Check if loading symbol disappears

       Severity: 1

    c. Instructions
        i. After completing Manual Test #1, the user reconnects to wifi after 15-30 seconds.

    d. Expected Result
        i. The loading symbol should disappear after reconnection on the web client.

3. **Manual Test #3**
    a. Identification and Classification

       Test Case 47

       System: loading.js Phase: 3

       Check if the loading symbol has motion

       Severity: 1

    b. Instructions
        i. The developer stimulates a situation (e.g. poor network situation) where the loading symbol appears on the client waiting for a response on the web and mobile app.

    c. Expected Result

        i. The loading symbol should be in motion to indicate the process is still occuring on the web client.

4. **Manual Test #4**
    a. Identification and Classification

       Test Case 48

       System: loading.js   Phase: 3

       Check if the user cannot navigate between pages when disconnected from wifi.

       Severity: 1
    a. Instructions
        i. The developer stimulates a situation (e.g. poor network situation) where the loading symbol appears on the client waiting for a response on the web and mobile app.
    b. Expected Result
        i. The web client should not allow the user to navigate between screens while disconnected from wifi, and they should see the loading symbol.

User Story #14

As a user, I would like to be able to set my profile picture.

1. **Manual Test #1**
    a. Identification and Classification

        Test Case 49

        System: camera.js, profile.html Phase: 3

        Check if user can choose to set or update profile picture

        Severity: 2

    a. Instructions
        i. A user clicks the profile image on the web client and chooses to set or update the profile picture.

    b. Expected Result
        i. The web client will successfully update the profile picture to the new profile picture selected.

2. **Manual Test #2**
    a. Identification and Classification

        Test Case 50

        System: camera.js, profile.html Phase: 3

        Check if user can upload pre-existing photo

        Severity: 2

    a. Instructions
        i. A user clicks the profile image on the web client and chooses to upload a pre-existing photo.

    b. Expected Result
        i. The web client will successfully update the profile picture to the new profile picture selected.

3. **Manual Test #3**
    a. Identification and Classification

        Test Case 51

        System: camera.js, profile.html Phase: 3

        Check if profile picture is displayed at the top of the web profile page

        Severity: 2

    a. Instructions
        i. A user has attempted to set their profile picture on the web app.

    b. Expected Result
        i. The web client will show a new profile picture on the top of the profile page.

4. **Manual Test #4**
    a. Identification and Classification
    Test Case 52
    System: camera.js, profile.html Phase: 3
    Check if the picture is cropped to fit the profile pic icon on the profile page.
    Severity: 2
    a. Instructions
        i. A user has attempted to set their profile picture on the web app.
    b. Expected Result
        i. The web client should crop the profile picture select to fit the icon on the profile page.

User Story #15

As a user, I would like to receive a confirmation email when I go to reset my password.

1. **Manual Test #1**
    a. Identification and Classification
    Test Case 53
    System: ChangePassword.js Phase: 1
    Check if the user is prompted to input their username and password when resetting their password.
    Severity: 1
    b. Instructions
        i. The user is on the web/mobile client and chooses to reset their password.
    c. Expected Result
        i. The web/mobile client should prompt the user to enter a username and password.
2. **Manual Test #2**
    a. Identification and Classification
    Test Case 54
    System: ChangePassword.js Phase: 1
    Check if the user inputs the correct username and password for resetting a password.
    Severity: 1
    b. Instructions
        i. The user is on the web/mobile client, chooses to reset their password, and enters their username and password.
    c. Expected Result
        i. The web/mobile client should call the backend route to verify the username and password match.
3. **Manual Test #3**
    a. Identification and Classification
    Test Case 55
    System: ChangePassword.js Phase: 1
    Check if the user receives an email with a unique 6-digit code
    Severity: 1
    b. Instructions
        i. The user is on the web/mobile client, chooses to reset their password, and enters their username and password.

c. Expected Result

   i. The user should receive an email with a unique 6-digit code and will be redirected to a page to input this code.

4. **Manual Test #4**

   a. Identification and Classification

      Test Case 56

      System: ChangePassword.js Phase: 1

      Check if the user is redirected to a page to input their new password.

      Severity: 1

   b. Instructions

      i. The user is on the web/mobile client, chooses to reset their password, enters their username and password, and inputs a unique 6-digit code correctly.

   c. Expected Result

      i. The user will be directed to a page to input their new password on the web client.