Tests and Test Plan for Illo

# Test Cases

# Use Case Testing

# Written Test 1

**Title: Confirm Permission Changes**

Actors: User

Requirements: Illo is in Permission Management

Main Scenario:

1. User selects “Save Changes” Button

2. Illo Saves the permission changes and returns to the Settings screen

Alternatives:

1a. User selects the “Cancel Changes” Button

1a1. Illo does not save the permission changes and returns to the Settings Screen

# Written Test 2

**Title: Start Internal Timer**

Actors: User

Requirements: Illo is on the Timer Screen

Main Scenario:

1. User selects the “Play” button.

2. Illo starts the current workout interval.

3. Play Button is Replaced with a Pause button.

# Written Test 3

**Title: Exit Permission Management**

Actors: User

Requirements: Illo is in Permission Management

Main Scenario:

1. User Selects the “Exit” Button.

2. Illo returns to the Setting Screen

Alternatives:

1a. User has unsaved changes

1a1. Illo displays a “Confirm Changes” popup.

# Written Test 4

**Title: Toggle Permission**

Actors: User

Requirements: Illo is in Permission Management

Main Scenario:

1. User toggles an enabled permission

2. Illo disables the selected permission

Alternatives:

1a. User toggles a disabled permission

1a1. Illo enables the selected permission

# Written Test 5

**Title: Rename Exercise from Activity Source**

Actors: User  
Requirements: Illo is in the Activity Source Configuration Screen

Main Scenario

1. User selects name of the Activity Source

2. Illo prompts the user to enter text

3. User enters text and confirms

4. Illo renames the Activity Source to the text entered

5. Illo returns to the Activity Source Configuration Screen

Alternatives:

1a. User exits the screen

1a1. Illo returns to the Activity Sources screen

3a. User cancels

3a1. Illo does not rename the Activity Source

# Written Test 6

**Title: Set Activity Source**

Actors: User

Requirements: Illo is in the Configuration Screen

Main Scenario:

1. User selects the Activity Source field

2. Illo swaps to Activity Sources screen

3. User selects an Activity Source

4. Illo presents list of exercises in selected Activity Source

5. User selects confirm for selected Exercise Set

6. Illo saves the selected Activity Source, and sets the timer’s nextExercise() method to point to thenext() method in the Activity Source

7. Illo closes dialogue and returns to the Illo Configuration Screen

Alternatives

3a. User exits screen

3a1. Illo returns to the Configuration Screen

5a. User selects cancel

5a1. Illo exits the confirmation menu, returning to the Activity Sources screen.

# Unit Testing

# Coded Test 1

# Coded Test 2

# Coded Test 3

# Coded Test 4

# Coded Test 5

# Alarm Activates Test

|  |  |
| --- | --- |
| Test Inputs | Expected Outcome |
| Timer Running | No Alarm |
| Timer Hits 0 and resets | Alarm runs once then stops |
| Timer is 0 | Alarm runs once then stops |

This test will make sure that the sound alarm will not interfere with the user’s manual input of timer duration.

To create this test:

1. separate test timer with varying lengths, (100sec, 10sec, 0sec)

2. Assertions that the alarm plays when the timer hits 0,

3. Assertion the alarm does not continue to run after once.

This test will run using the override functions in the mainActivity of the app.

# UI color Swapping Test

|  |  |
| --- | --- |
| Test Inputs | Expected Outcome |
| Default | All windows are white bg, blue border |
| Green | All windows are light blue bg, green border |
| Red | All windows are light red bg, red border |

This test will make sure that all UI windows are affected by changing the color options.

1. Each window will need to be checked for correct color

2. Each window will need to be checked that they stay the selected color when app is closed

This test will run using the window swapping functions in the mainActivity of the app as well as the color saving controller.

# Username Input Test

|  |  |
| --- | --- |
| Valid | Invalid |
| 1 < Length < 32 | Length = 0, Length> 32 |
| Includes Characters | No Special Characters (Except Space) |
| Can Include Numbers | No Emojis |
|  | Crass Words |

This test will make sure that user input of username will be viable with motivational text. This test will also do a rudimentary check for crass/inappropriate words.

1. Input will need to be checked that it is saved to the app

2. Input must have a length of at least 1 and at most 32.

3. Input must not use special characters

4. Input must not use Emojis

5. Input will be checked for inappropriate words and reject if found.

This test will run using the profile class and a special username checker class.

# Streak Test

|  |  |
| --- | --- |
| Test Inputs | Expected Outcome |
| 24 hours pass | Reset Streak |
| A Set is completed | Streak +1 |
| A Set is completed after a set has been completed | No change |

This test will check if the user use streak is accurate to usage daily.

1. Check last time the app was opened, if 24 hours streak should be 0

2. Check if last time streak was increased, if 24 hours streak should be 0

3. Check if last time streak was increased, if less than 24 hours + set complete Streak +1

4. Check if Streak has been increased in 24 hours, + set completed, Streak + 0

This test will run using the Set Class, and the Set controller as well as the profile class.

# Feedback Test

|  |  |
| --- | --- |
| Test Inputs | Expected Outcome |
| Empty Report | Will not Accept |
| Rating | Save Ratings to rating Database |
| Rating + Text | Save Ratings to rating Database & Text to Text Database |

This test will make sure that user Feedback will be saved accurately in an external Database.

1. If the report is empty, the system should not allow submission

2. If the report is just a rating, check if it is sent to the correct rating Database

3. If the report is a rating and text, check if it is sent to the correct rating and text databases.

This test will run using the Feedback Controller, and Rating and Text Databases.