

Functions

[update\(\)](#)

In order to update the value of a stat, for instance distance, we need to update that value every couple milliseconds. We assign the values to the specific text attributes of each stat.

[startExerciseButton\(\)](#)

Checks if the 'start' button has been activated and starts the 'run' exercise plus activates gps tracking. Should the user not run anymore, autopause is activated automatically.

[stopExerciseButton\(\)](#)

Stops the running exercise and goes back to the main screen

[resumeExerciseButton\(\)](#)

If the runner decides to keep moving, we resume the exercise after a click on this button

[watchPosition\(\)](#)

Watches the actual position of the runner while running and provides longitude and latitude for further usage

[onstatechange\(\)](#)

if the state of our exercise changes, we fire up this function and check in what state we are in. depending on the state, we show and hide specific views and buttons

update()

In order to update the value of a stat, for instance distance, we need to update that value every couple milliseconds. We assign the values to the specific text attributes of each stat.

Kind: global function

startExerciseButton()

Checks if the 'start' button has been activated and starts the 'run' exercise plus activates gps tracking. Should the user not run anymore, autopause is activated automatically.

Kind: global function

stopExerciseButton()

Stops the running exercise and goes back to the main screen

Kind: global function

resumeExerciseButton()

If the runner decides to keep moving, we resume the exercise after a click on this button

Kind: global function

watchPosition()

Watches the actual position of the runner while running and provides longitude and latitude for further usage

Kind: global function

onstatechange()

if the state of our exercise changes, we fire up this function and check in what state we are in. depending on the state, we show and hide specific views and buttons

Kind: global function