

# Remote Command Service, ReCoS

---

Remote Command Service kurz ReCoS ist ein Programm, um Apps, Programme und Skripte aus der Ferne, von einem anderen Gerät aus, auszuführen zu können. Genau wie eine Makrotastatur, aber mit visuellem Feedback und mehr Interaktion. Du kannst mit deinem Handy oder Tablett auf deinem Rechner Programme starten, Tastatur Makros ausführen oder dir einfach nur die Uhrzeit anzeigen lassen.

Das Programm besteht aus 3 verschiedenen Komponenten.

Zunächst ist da das Hauptprogramm, ein kleiner Golang-basierter Mikroservice, der alle Funktionen auf dem zu steuernden Rechner bereitstellt. Dieser Dienst läuft gewöhnlich im Hintergrund und sollte automatisch mit dem Rechner gestartet werden.

Auf der anderen Seite gibt es zwei Webanwendungen.

Einmal der Client, der die Aktionen steuert, der auf dem steuernden Gerät gestartet werden kann, und ein Client der für die Konfiguration zuständig ist.

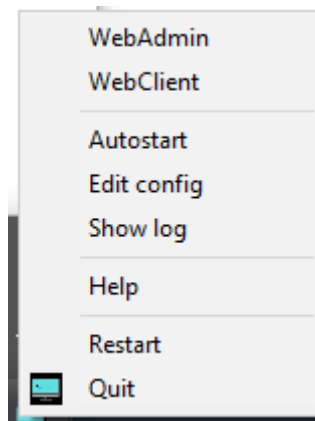
Folgende Funktionen können gesteuert werden: (Das ist natürlich nur ein Auszug aus den wichtigsten Funktionen)

- Lautstärke und Mikrofon Gain.
- OBS Studio integration: Steure Profile, Szenen und Stream und Aufnahme Funktionen
- Starte einen Browser, den Explorer oder irgendein anderes Programm auf Tastendruck
- zeige dir ein Uhr in verschiedenen Designs an
- Zähle Ereignisse auf Knopfdruck
- zähle die Tage bis zu einem Ereigniss
- lass die Würfel rollen
- zeige dir Sensordaten deines PC an, CPU/GPU Auslastung oder Temperatur
- Kontrolliere deine Smarthome Produkte wie Homematic(C) oder Philips Hue(C)
- benutze die virtuelle Tastatur und sende beliebige Tastenfolgen
- steuere deinen Media Player
- zeig dir die Netzwerk Ping Zeit zu einem anderen Server an
- mach dein eigenes Soundboard: spiele beliebige Mediadateien ab
- zufalls Wörter: lass dir aus einer Liste von Wörter zufällig eines auswählen
- mache einen Screenshot und speichere diesen automatisch in einen Ordner deiner Wahl
- zeige dir eine Stoppuhr an
- benutze einen Countdown-Timer
- Kontrolliere deinen Windows Desktop, aktiviere, verschiebe und minimiere Fenster

## Installation

---

Die Installation ist einfach. Füh das Installationsprogramm aus. Nach der Installation startet der Dienst automatisch. Für den automatischen Start musst du nur im Kontextmenü Autostart wählen.



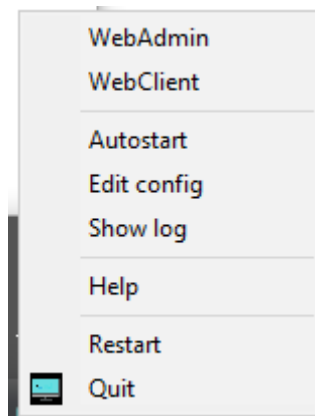
Jetzt kannst du einfach den Admin-Client mit WebAdmin aufrufen.

Für den Client wählst du WebClient. Wenn du den Client auf einem anderen Gerät ausführen möchtest, schau im nächsten Kapitel.

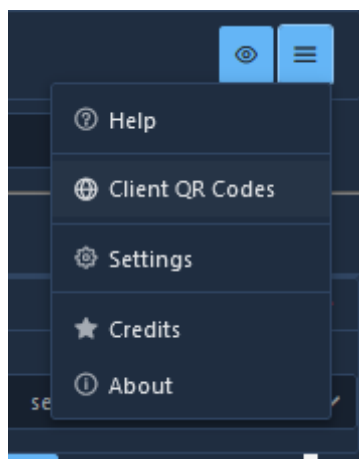
## ReCoS Client - Web Client Interface

### Verbinde ein Gerät

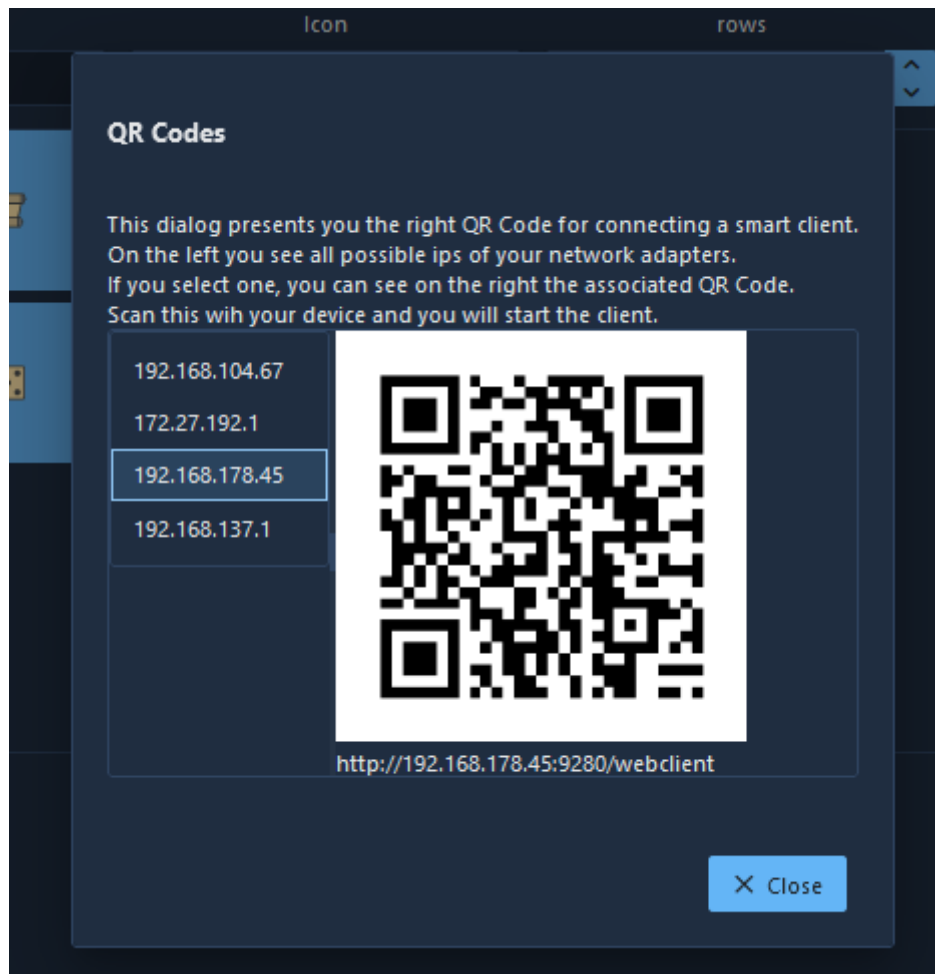
Einen Client zu verbinden ist einfach. Um den Client direkt auf derselben Maschine wie den ReCoS Service zu starten, klicke einfach auf das Task-Icon-Menü für den WebClient.



Wenn Du ein anderes Gerät, wie dein Smartphone oder Tablett, verbinden möchten, musst du dich zunächst mit dem Gerät verbinden, auf dem der ReCoS-Dienst ausgeführt wird, vorausgesetzt beide Geräte sind im gleichen Netzwerk. Um diesen Vorgang zu vereinfachen, kann dir ReCoS fertige QR Codes für die Verbindung generieren. Dazu starte den Webadmin über das obige Menü. Klicke auf das Hilfemenü und dann auf das Menü Client QR Codes.



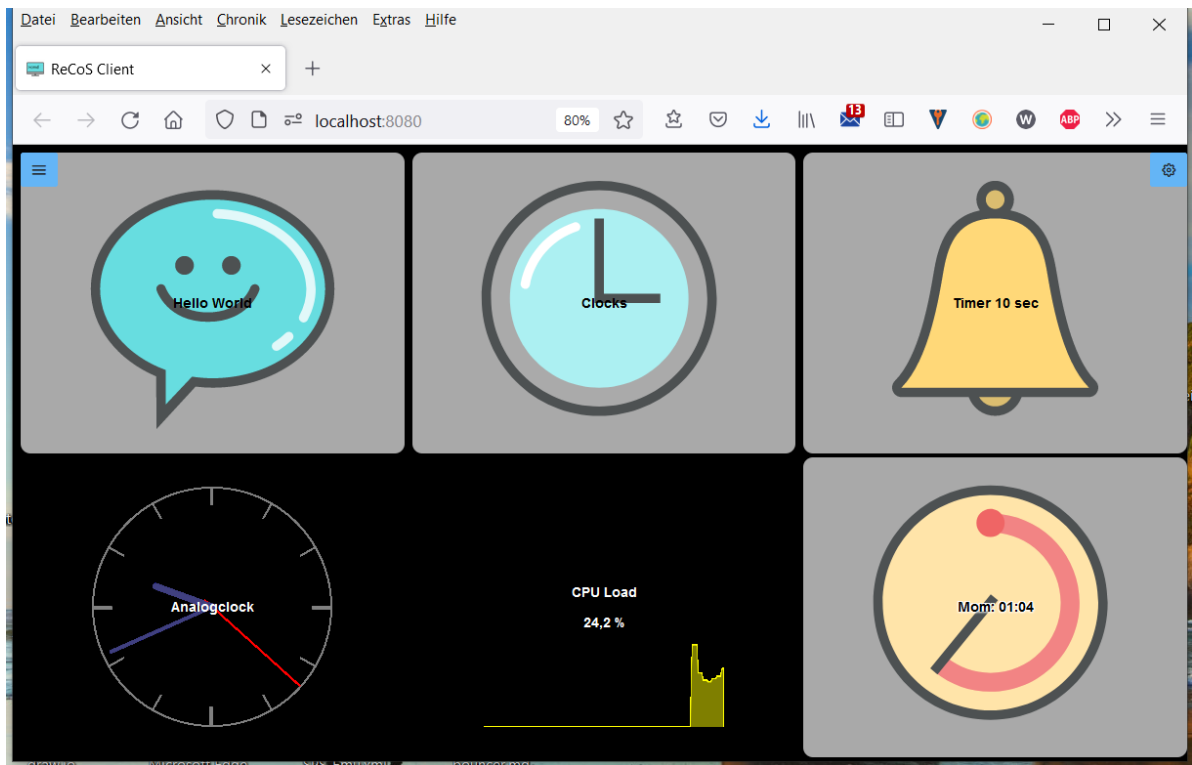
Wenn dieser Dialog sichtbar ist, kannst du eine Netzwerkschnittstellen auswählen und den QR-Code dafür scannen. (Falls du keine Möglichkeit hast, einen QR Code auf dem Gerät zu scannen, kannst du auch einfach den Interbrowser auf dem Gerät starten und die Adresse, die unter dem QR Code steht, verwenden.) Nachdem du diesen Code ausgeführt hast, wirst du zur richtigen Webseite weitergeleitet. Wenn es ein Problem gibt, verwende einfach einen anderen Eintrag. (Nicht jede Schnittstelle ist mit deinem Heimnetzwerk verbunden. Einige sind nur für den internen Gebrauch. Das Programm kann aber leider den Unterschied nicht erkennen.)



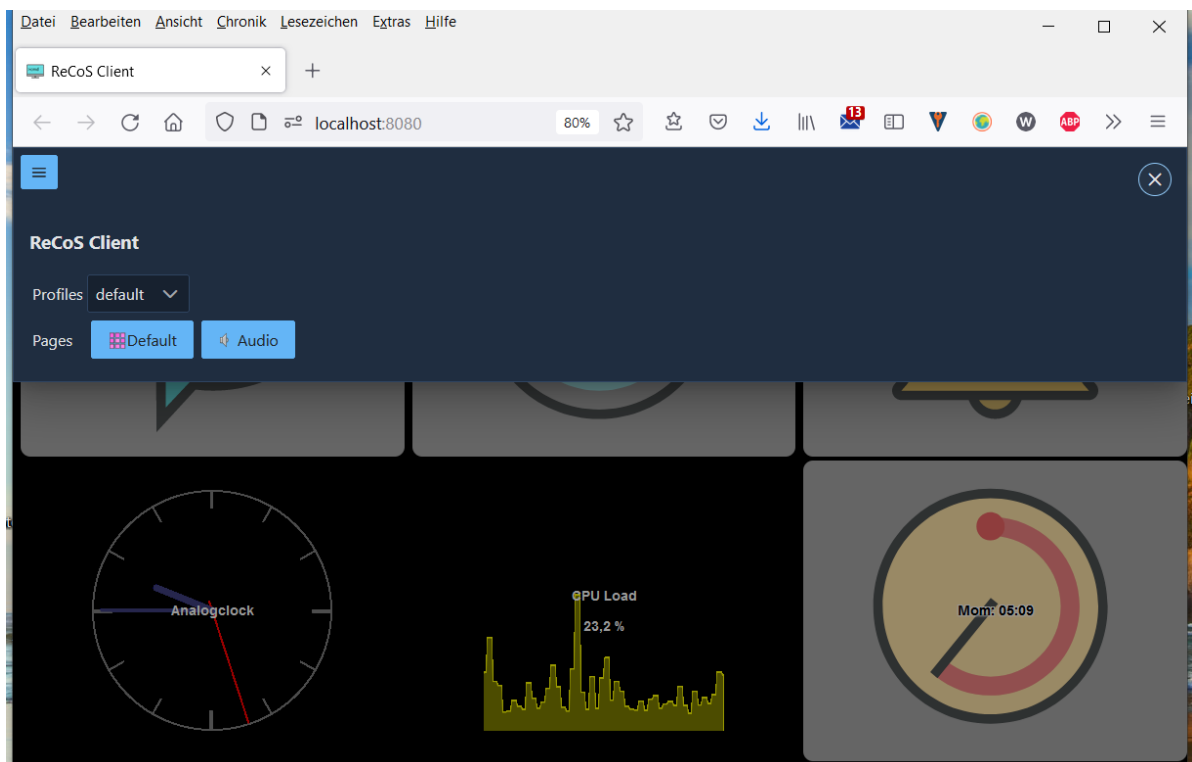
## Der WebClient

---

Dies ist der Haupt-ReCoS-Webclient zum Ausführen der konfigurierten Befehle.



Der Client hat einen großen Schaltflächenbereich, der die verschiedenen Aktionen präsentiert. Aber fangen wir ganz am Anfang an. Zunächst ist alles in einem **Profil** organisiert. Du kannst unterschiedliche Profile für unterschiedliche Clients oder unterschiedliche Szenarien anlegen. Ein Client kann jeweils nur ein Profil präsentieren, aber du kannst verschiedene Profile gleichzeitig in verschiedenen Browsern geöffnet haben. Beispiel: Du kannst dir ein spezielles Profil für dein Telefon, ein Anderes für den Tablett und ein wirklich großes Profile für deinen PC anlegen. Und alle können gleichzeitig aktiv sein. Aber um es noch einmal zu betonen, Du kannst das gleiche Profil auch auf 2 verschiedenen Geräten gleichzeitig betreiben. Beim Starten des Clients wird immer das erste Profil ausgewählt. Du kannst das Profile mit der Profil-Combobox auf der Einstellungsseite auswählen. Zum Öffnen einfach den Button in der oberen rechten Ecke drücken.



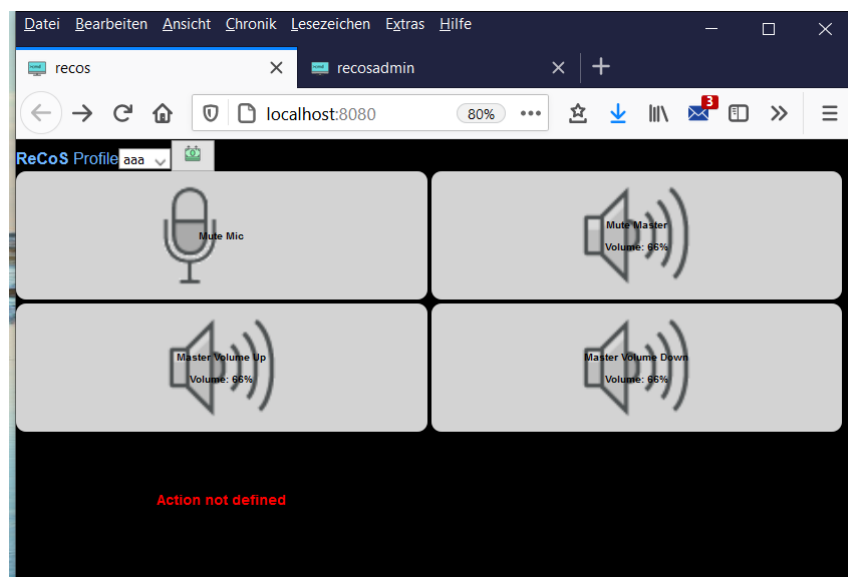
Jedes Profil hat unterschiedliche **Seiten** zur weiteren Strukturierung. Beim Starten des Clients wird automatisch die erste Seite ausgewählt. Eine Seite besteht dann aus Zeilen und Spalten. Du kannst die Größe der Seite im Admin-Client einstellen. Je mehr Zeilen und Spalten du hast, desto kleiner werden die einzelnen Aktionen. Die Seite passt ihre Aktionen dem möglichen Raum an. Seiten können direkt über die Einstellungen (dargestellt als Schaltflächen, im Bild oben Default und Audio) oder über spezielle Aktionen geändert werden. Ob eine Seite im Einstellungsdialog erscheint, kann in der Seitenkonfiguration eingestellt werden. Wenn die Seite kein Symbol hat, wird nur der Name angezeigt.

Im Client-Bereich siehst du 2 Arten von Schaltflächen und möglicherweise etwas Leerraum. Zunächst sind da die grau hinterlegten Tasten, die Du drücken kannst. (**Single Action**) Dies ist das Standardverhalten. Beim Drücken wird die zugrunde liegenden Aktion auf dem Computer ausgeführt.

Die schwarz hinterlegten Schaltflächen dienen nur zur Anzeige. (**Display Action**) Dort werden Systemzustände oder andere Parameter angezeigt.

Die leeren Felder warten darauf, das Du dort eine Aktion zuordnest.

(Wenn Du ein Feld mit der roten Warnung „Action not defined“ siehst, liegt eine Fehlkonfiguration vor. Möglicherweise hast Du eine Aktion gelöscht und vergessen diese von der Schaltfläche zu entfernen.)



Schaltflächen oder Aktionen können mehrere Zustände haben. Je nach Situation werden dort dann unterschiedliche Icons angezeigt. Wenn ein Befehl aktuell ausgeführt wird, wird normalerweise die Sanduhr angezeigt. Bei mehreren Kommandos pro Aktion zeigt die Stellung der Sanduhr an, wie viele Befehle gerade ausgeführt worden sind. (Ja, Aktionen können eine Befehlsliste enthalten)

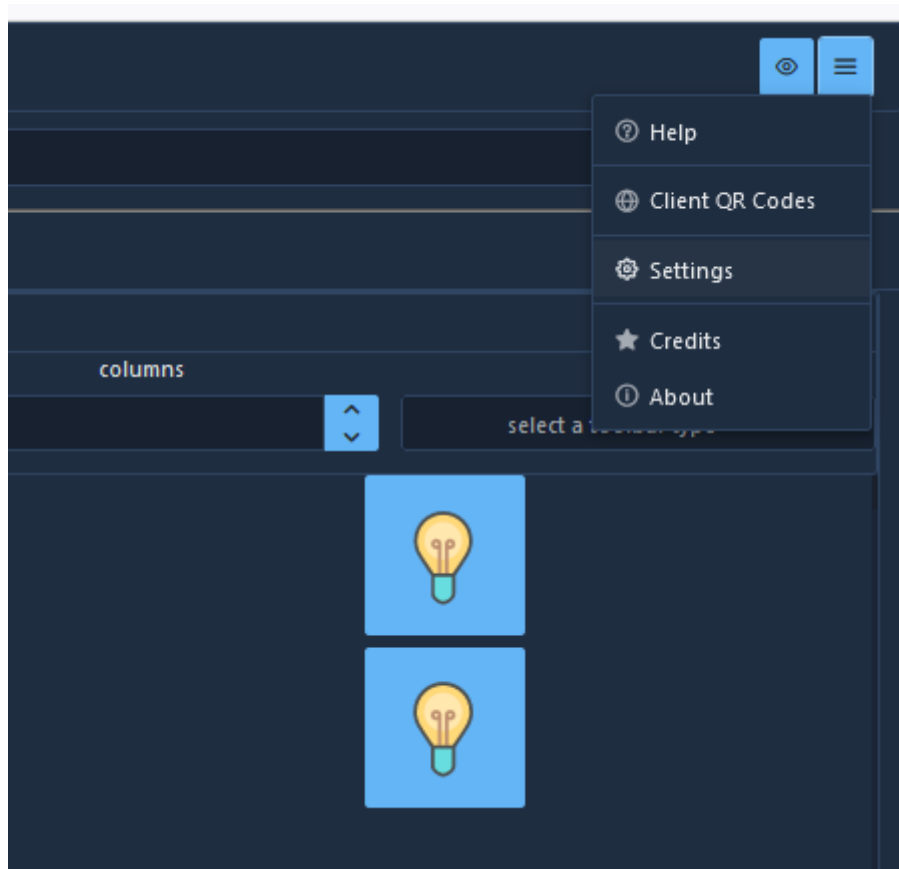
Besondere Aktionen: Bei sogenannten **Multi**-Aktionen durchläuft die Aktion eine Liste von anderen Aktionen. Die 1. Aktion wird beim 1. Drücken ausgeführt, die 2. beim 2. Drücken und so weiter. Der jeweilige Status wird durch das Symbol der entsprechenden Aktion dargestellt.

Wie bereits erwähnt, kann eine Aktion mehrere Befehle enthalten. (Die Aktion "Hello World" von der Demo enthält z.B. 3 Befehle: Notepad starten, einige Sekunden warten, "Hello ReCoS" schreiben)

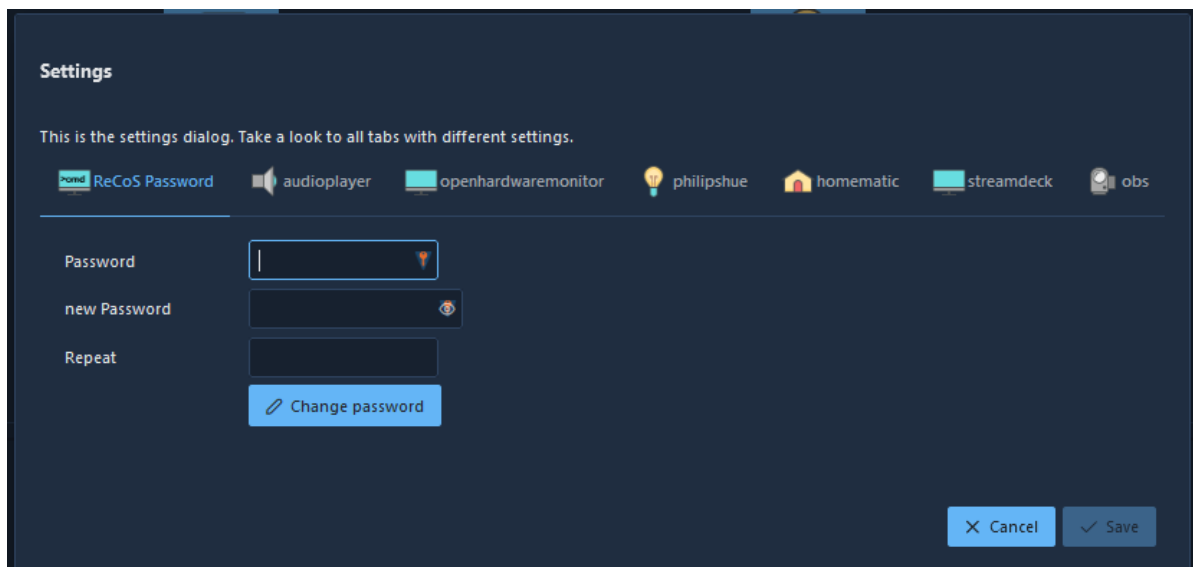
## ReCoS Admin - Die Konfiguration

Der WebAdmin dient zur Konfiguration des ReCoS Systems.

Alle modifizierenden Funktionen sind mit einem Passwort geschützt. Dieses Passwort kannst du im WebAdmin festlegen. Das Standardpasswort ist `recosadmin`. Zum Ändern gehst du auf das Icon rechts oben und wählst Settings




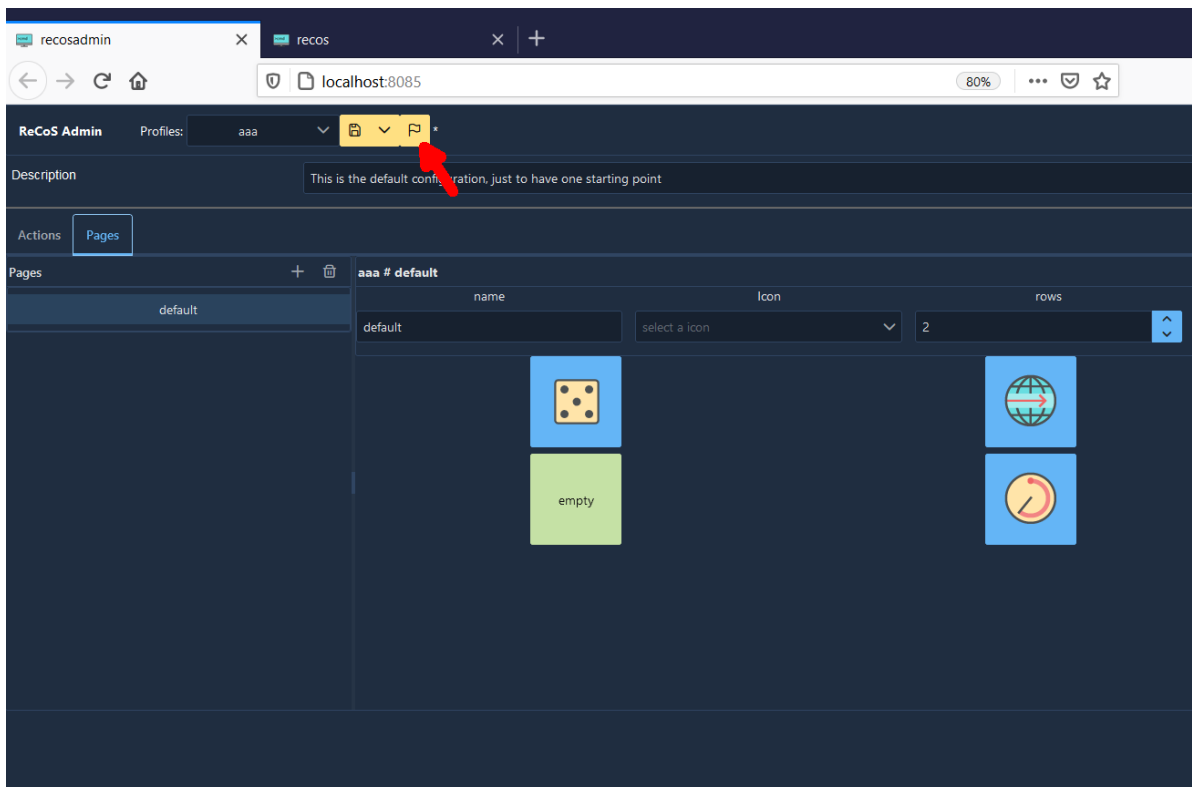
Im folgende Dialog findest du die Passwordeinstellung ubter dem Reiter ReCoS Password



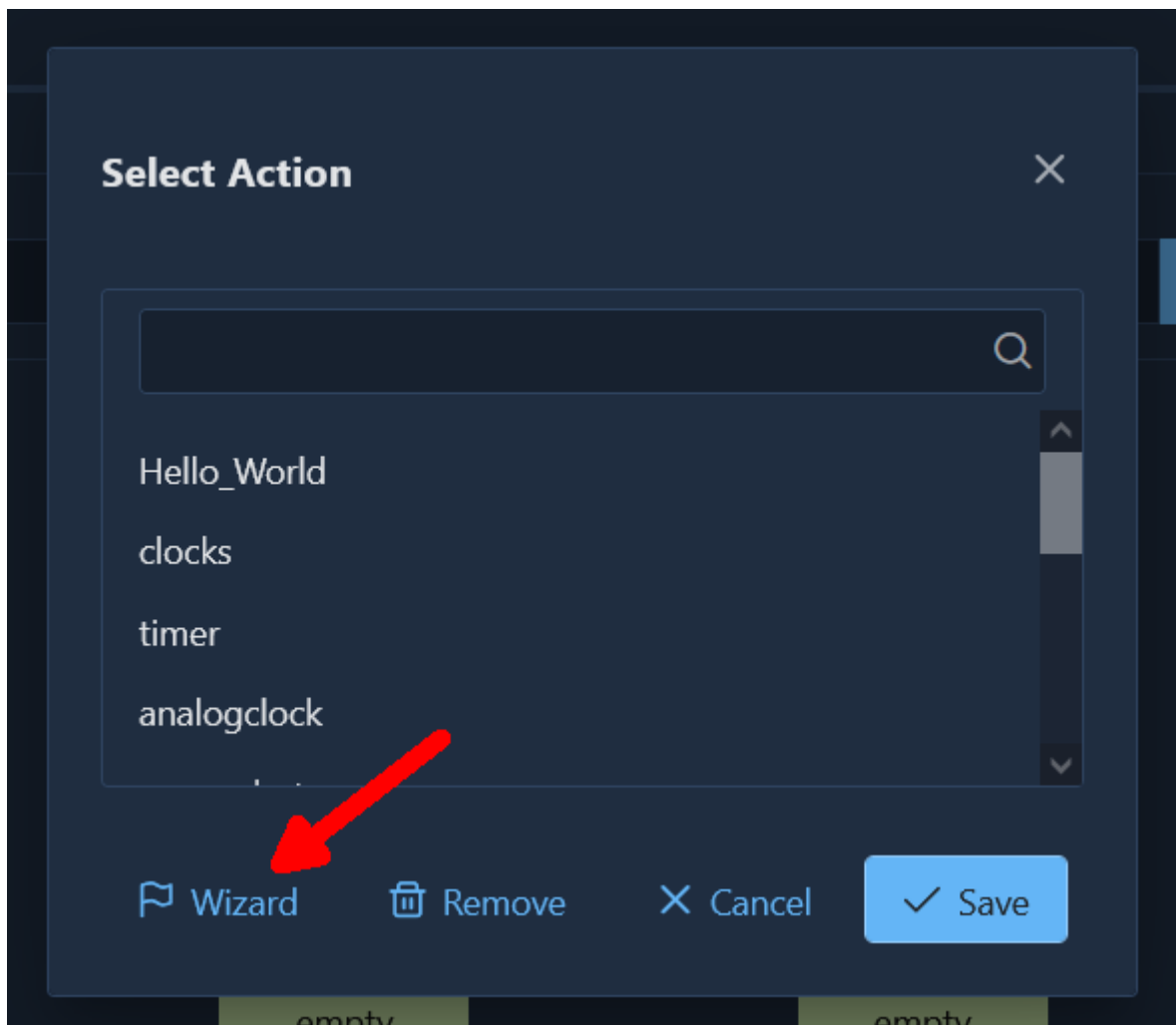
Wie üblich musst du hier sowohl dein altes Passwort wie auch 2x dein neues Passwort eingeben. Um die Passwortprüfung zu deaktivieren, benutze einfach ein leeres Passwort.

## Action Wizard

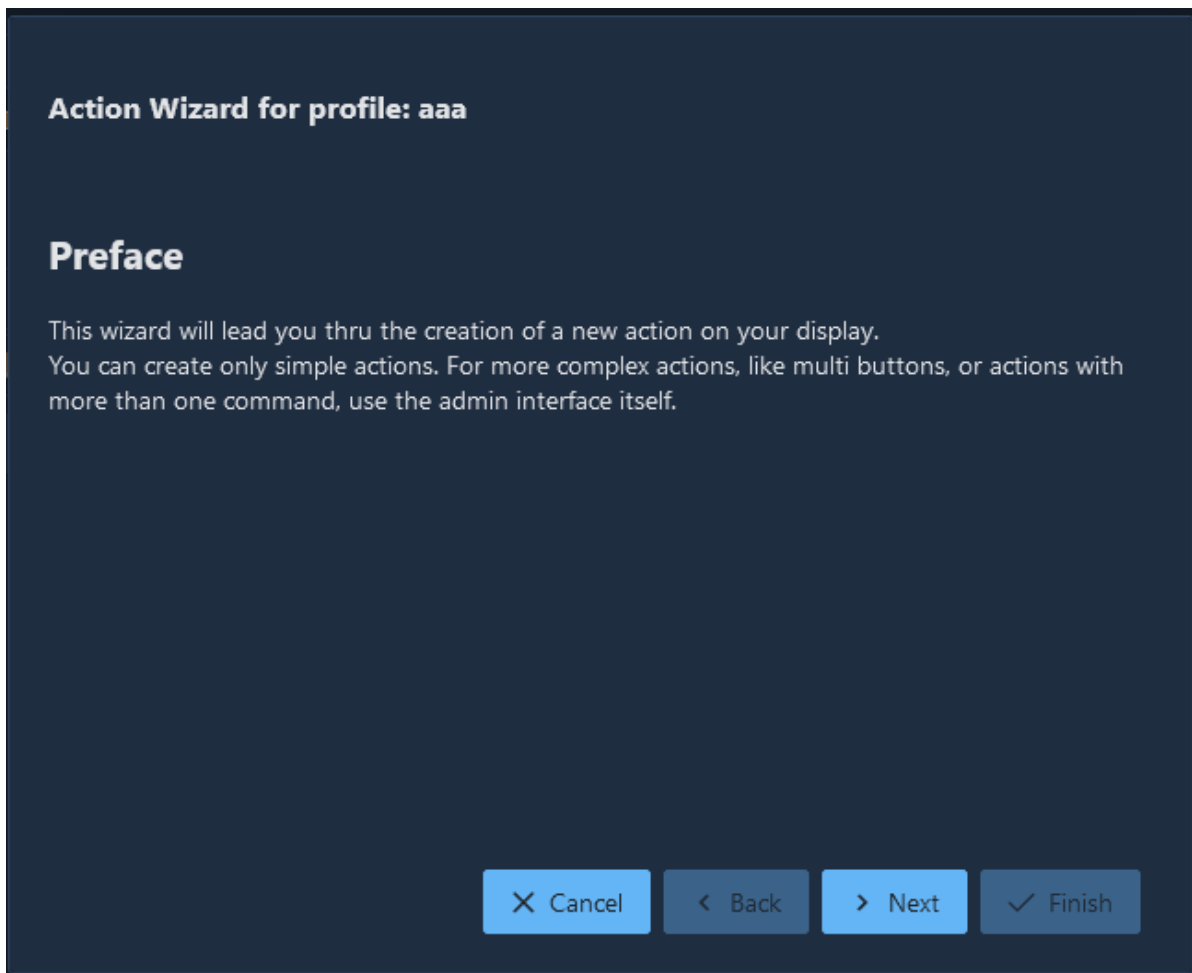
Mit dem Aktionsassistenten könntst du schnell und einfach neue Aktionen erstellen. Mit dieser Schaltfläche  starten Sie den Aktionsassistenten.



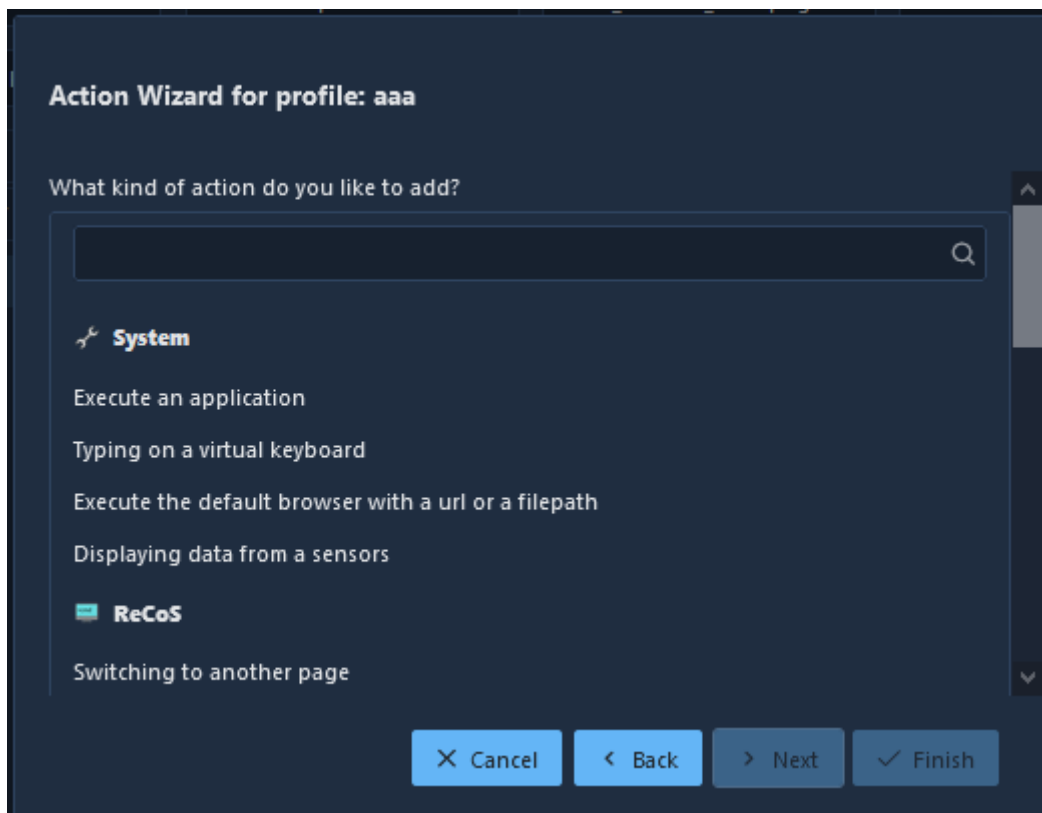
Oder im Schaltflächenzuordnungsdialog:



Zuerst siehst du diese Vorwortseite:



Auf der nächsten Seite kannst du den Befehl auswählen, der generiert werden soll. Du kannst das Suchfeld verwenden, um die Liste der Befehle zu filtern.



Weiter mit Next.

Im nächsten Dialog kannst du einige Angaben zum Befehl machen. Du kannst der Schaltfläche einen Titel geben, ein Symbol zuweisen (eine Vorgabe wird direkt durch den Befehl vorgegeben) und du kannst je nach Befehl verschiedene andere Einstellungen vornehmen. Weiter mit Next.



### Action Wizard for profile: aaa

Set the parameters for Timer

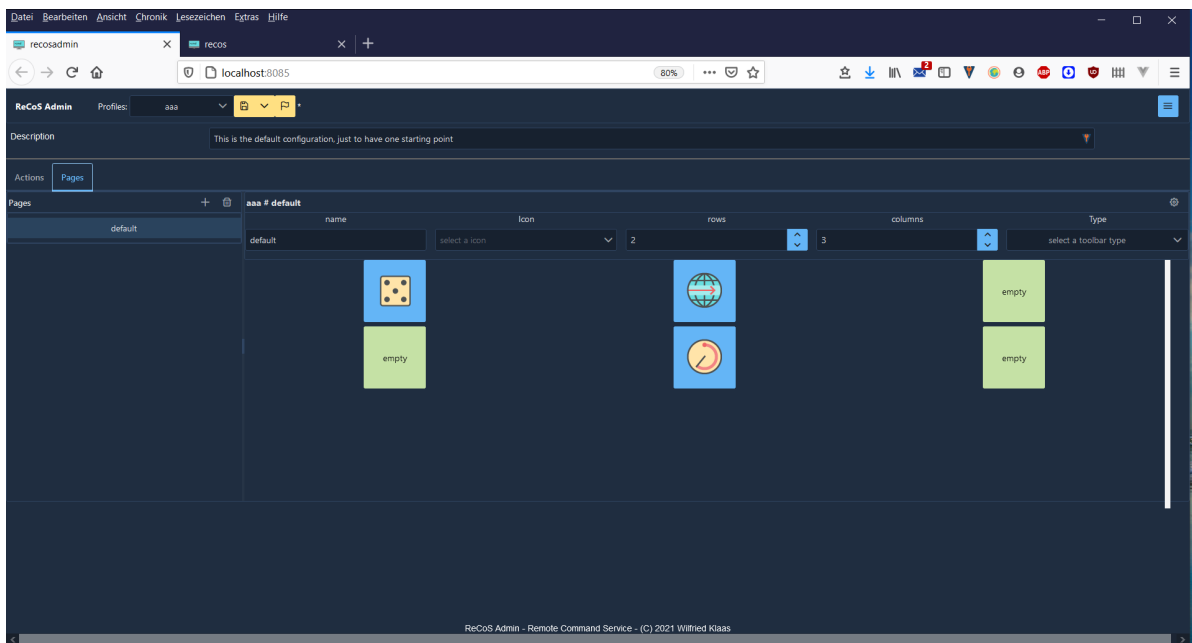
Title	<input type="text"/>
Icon	<input type="text" value="timer.png"/>
time	<input type="text"/> <div>^ v</div>
finished	<input type="text"/>

### Action Wizard for profile: aaa

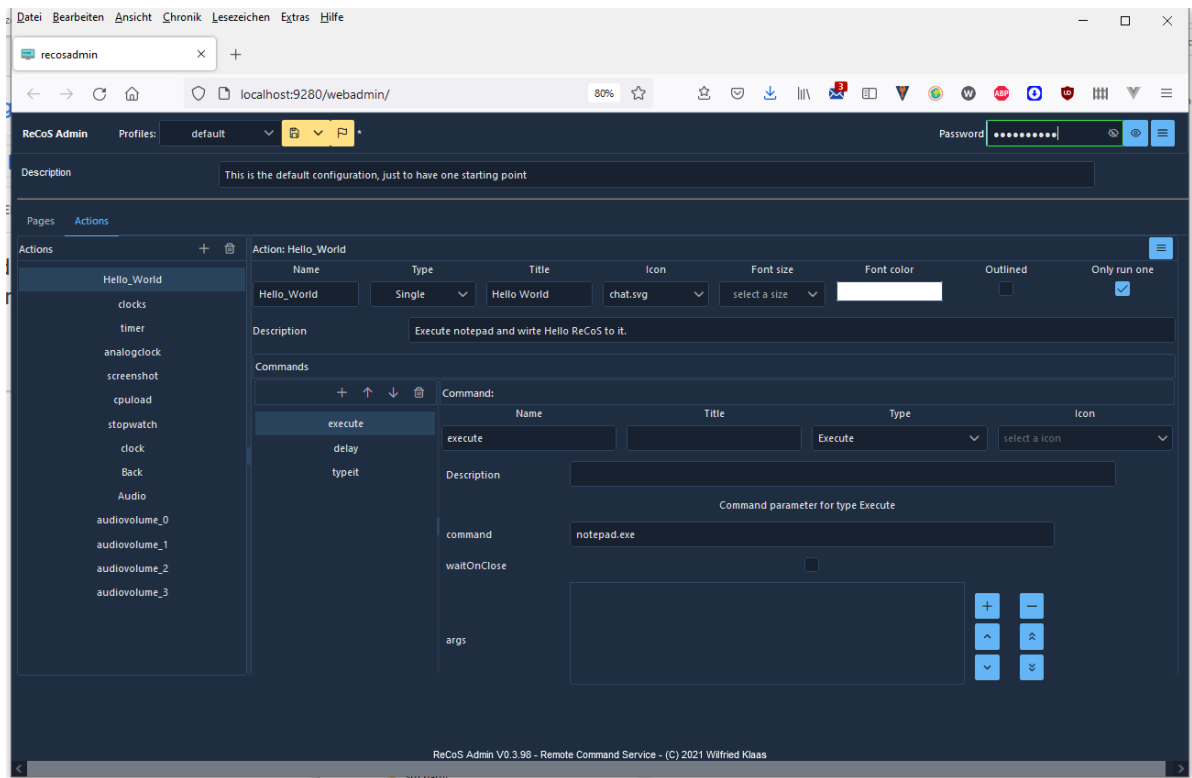
Set the parameters for Timer

Title	<input type="text" value="Count down"/>
Icon	<input type="text" value="timer.png"/>
time	<input type="text" value="60 Seconds"/> <div>^ v</div>
finished	<input type="text" value="Ready"/>

Auf dieser letzten Seite kannst du festlegen, wo die Aktion angezeigt werden soll. Du kannst hier die bereits vorhandenen Seiten auswählen oder eine neue Seite erstellen. Klicke einfach auf einen freien Button, auf dem die Aktion gespeichert werden soll. Mit Finish wird jetzt die Seite und das Profil gespeichert.



Im Admin Client im Bereich Actions siehst du dann, was der Assistent generiert hat.

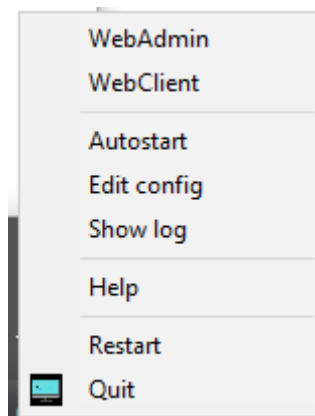


## Austauschen von Profilen, Aktionen und konfigurierten Seiten

Ihr könnt sowohl ganze Profile, Seiten wie auch einzelne Aktionen exportieren und auch wieder importieren. (z.B. um Sie mit Freunden zu teilen) Bei verschiedenen Aktionen müsst ihr euch aber versichern, dass die Aktionen nach einem Import noch funktionieren. Bei dem Kommando "Programm ausführen" steht üblicherweise der Pfad zu der ausführbaren Datei in der Konfiguration. Dieses kann natürlich auf deinem System ein anderer sein als der auf dem System deines Freundes.

## ReCoS Service

Der Service ist der Hauptbestandteil des ReCoS. Der Service ist die Stelle, die all die schönen Dinge tut. Allerdings siehst du davon nur ein kleines Symbol in der Taskleiste. Und dort gibt es ein kleines Kontextmenü.



- **WebAdmin, WebClient:** Du kannst direkt das Webinterface oder den Admin-Client starten.
- **Autostart:** Du kannst den Dienst so registrieren, dass er beim Windows-Start automatisch gestartet wird.
- **Edit config:** Die nächste Option besteht darin, die Datei service.yaml zu bearbeiten, die die gesamte Konfiguration für den Dienst selbst enthält. Normalerweise brauchst du hier nichts

zu tun, aber nur für den Fall...

- **Show log**: Ähnliches gilt für das Log.
- Mit **Hilfe** bist du bestimmt hier hin gekommen.
- Und **Restart** startet den Service neu.
- **Quit**: Der letzte Menüeintrag dient zum Beenden des Dienstes.

## Profilkonfiguration

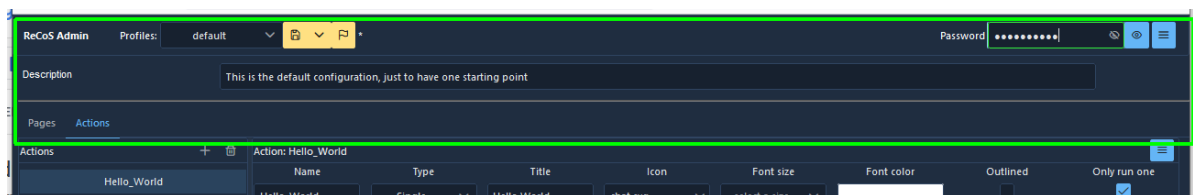
Im ReCoS können verschiedene Profile hinterlegt werden. Jedes Profil besteht zum einen aus einer Liste von Aktionen und zum anderen aus den Seiten, die diese Aktionen auf einer Oberfläche repräsentieren. Jedes Profil hat einen Namen und eine Liste von Aktionen und Seiten.

`name` : Der Name des Profils

`description` : eine vom Benutzer lesbare Beschreibung

`pages` : Dies sind die verschiedenen Seiten zum Anzeigen der Aktionen

`actions` : eine Liste verschiedener Aktionen. Eine Aktion kann auf verschiedenen Seiten erscheinen. Da es sich jedoch um dieselbe Aktion handelt, ist der Status/das Ergebnis gleich.



## Seiten

Eine Seite ist eine Ansichtskomponente, die mit Zeilen und Spalten angezeigt wird. Jede dieser Zellen visualisiert dann eine Aktion. Die Zellenliste wird über den Namen in der Aktionsliste mit einer Aktion verknüpft. Der Index einer Aktion einer Zelle wird berechnet als

$$\text{index} = (\text{cell.row} * \text{page.rows}) + \text{cell.column}$$

Parameter:

`name` : Der Name der Seite

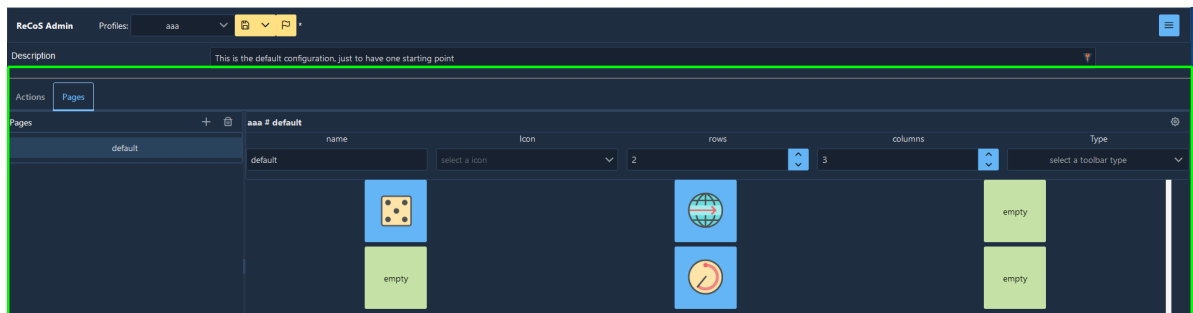
`icon` : das Icon dieser Seite

`rows` : Anzahl der Zeilen auf der Seite

`columns` : Anzahl der Spalten auf der Seite

`toolbar` : `show`, diese Seite erscheint in der Toolbar, um direkt zu wechseln, `hide`, wenn diese Seite in der Toolbar erscheinen soll. Mit dem Seitenbefehl können Sie auf diese Seite wechseln.

`cells` : Liste der Namen der Aktion pro Zelle (Im UI als Buttonfeld angezeigt)



# Aktionen

Eine Aktion ist der Teil, der definiert, was zu tun ist, wenn eine Schaltfläche ausgelöst wird.

Folgende Parameter werden verwendet:

**Name** : ist der Name der Aktion

**type** : **SINGLE** ist eine Einzelaktion. Die Aktion startet immer die Befehlsliste.

**DISPLAY** ist für die reine Anzeige. Es werden nur Text, Symbole oder Bilder angezeigt, Du kannst jedoch nicht damit interagieren.

**MULTI** ist die dritte Option. Hier kannst du mehrere Schritte definieren und auf jedem Schritt eine Aktion definieren, die beim Aktivieren dieses Schrittes ausgelöst wird.

**Title** : der Titel der Aktion, die von der Benutzeroberfläche verwendet wird

**Description** : benutzerdefinierte Beschreibung dieser Aktion

**Icon** : das Symbol, das auf der Zelle angezeigt wird

**FontSize** : die Größe des Titels und des Textes, standardmäßig 14

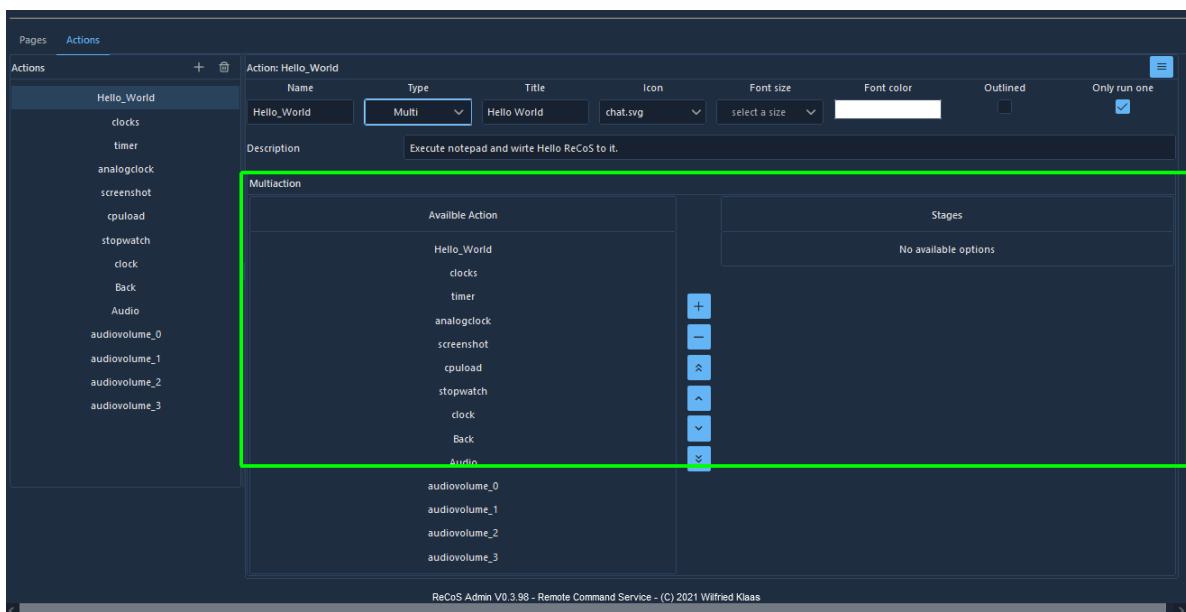
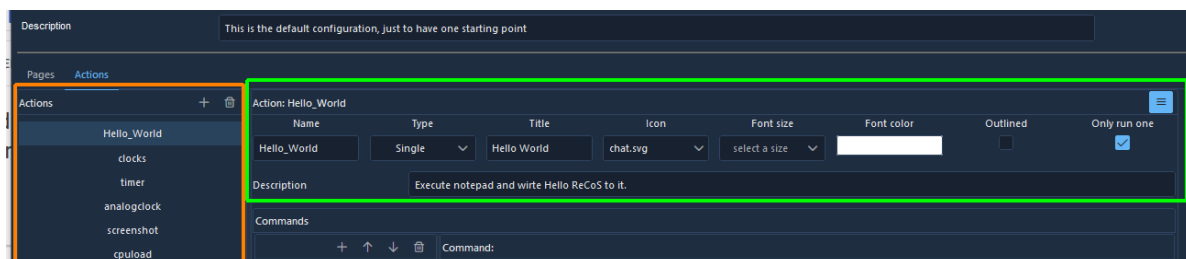
**Fontcolor** : die Farbe des Titels und des Textes, standardmäßig schwarz

**outlined** : wahr oder falsch, manchmal ist das Lesen eines schwarzen Textes auf schwarzem Grund etwas schwierig. Outline kann dort helfen.

**only run once** : ist wahr oder falsch. Wenn "runonce" wahr ist und die Aktion zweimal ausgelöst wird, müssen alle Befehle der ersten Ausführung beendet sein, bevor die zweite Ausführung stattfindet. Bei false wird die Ausführung direkt gestartet, ohne den vorherigen Ausführungsstatus der Aktion zu überprüfen. Evtl. laufen dann beide Ausführungen parallel.

**Commands** : Liste der Befehle zur Ausführung dieser Aktion

**Actions** : Gilt nur in einer MULTI-Aktion. Für jede Stufe sollte der Name der Aktion stehen, die aufgerufen wird, wenn die Stufe ausgeführt wird. Wird eine Stufe ausgeführt, wird das Icon der zuletzt ausgeführten Aktion (Stufe) als Icon der Multiaktion angezeigt und der Titel wird in der Textzeile angezeigt.



# Commands

Commands sind die Kommandos, also der Teil, der ausgeführt wird. Dies sind die Einstellungen, die in allen Befehlen gleich sind.

**Name** : benennt den Befehl

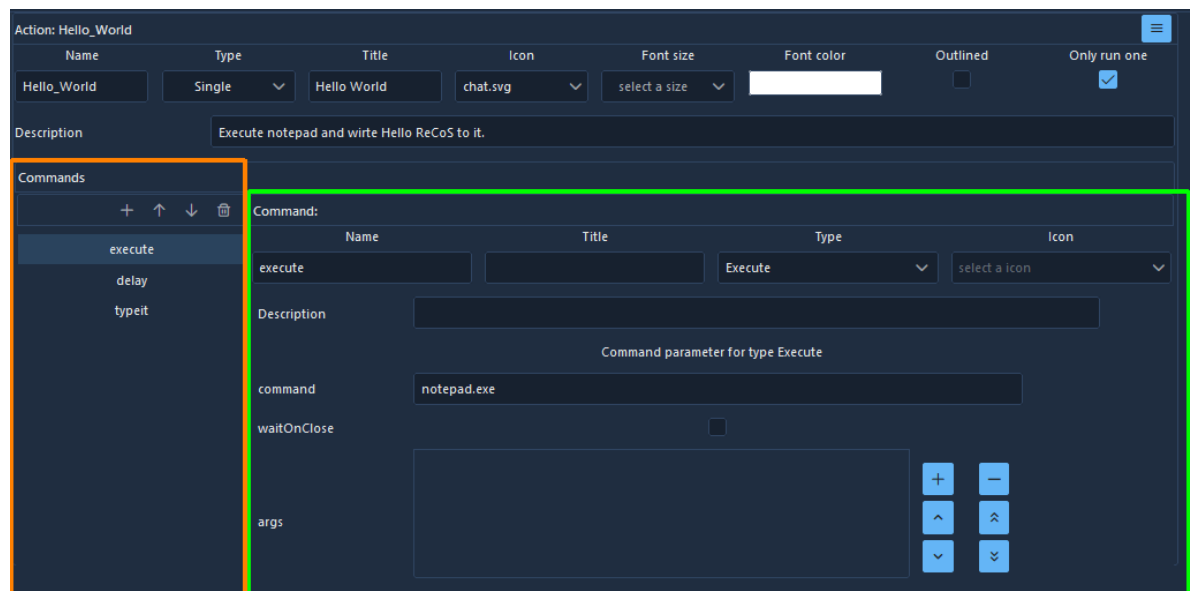
**Title** : ein Text, der angezeigt wird, wenn dieser Befehl ausgeführt wird

**Type** : der Typ des Befehls

**Icon** : ein Icon, das beim Ausführen dieses Befehls angezeigt wird

**Description** : eine vom Benutzer lesbare Beschreibung

Die anderen Parameter sind von Befehl zu Befehl verschieden.



## No Operation

Tut nichts. (Du kannst aber den Text und das Symbol ändern ;-)

**Type** : NOOP

**Parameter** : keine

## Audiocontrol

Übernimmt die Kontrolle über Ihre Audiogeräte.

Mit diesem Befehl können Sie verschiedene Audiogeräte steuern, Einstellen von Lautstärke oder Stummschaltung.

**type** : AUDIOVOLUME

**Parameter**:

**device** : das Gerät, das Sie steuern möchten. Es gibt verschiedene Geräte in Ihrem System.

Welche Sie auswählen können, sehen Sie in der Admin-Oberfläche. Es gibt 2 Voreinstellungen:

**master** für den Master-Ausgang. Dies ist bei Windows diejenige, die Sie direkt mit dem Taskleistensymbol steuern können. Und **mic**, welches das Standardeingabegerät darstellt.

**Befehl** : Dies ist der Befehl, den Sie abfeuern möchten. **mute**, um den Mute-Status umzuschalten. **volume up** zum Erhöhen und **volume down** zum Verringern der Lautstärke dieses Geräts.

## Clock



Nur eine Uhr in verschiedenen Designs.

Type: UHR

Parameter:

**format**: das Format der Uhrzeit in der Golang-Format-Syntax, Standardwerte: 15:04:05

**timezone**: String mit dem Namen der Zeitzone der IANA Timezone Database ([https://en.wikipedia.org/wiki/List\\_of\\_tz\\_database\\_time\\_zones](https://en.wikipedia.org/wiki/List_of_tz_database_time_zones)), die verwendet werden soll.

**dateformat**: das Format des Datums in der Golang-Format-Syntax, das von verschiedenen Designs verwendet wird, Standardwerte: 02.01

**graphical**: wahr oder falsch, zeigt eine analoge Uhr an

**showseconds**: Zeigt die Sekunden im Design an.

**showdate**: Zeigt das Datum im Design formatiert mit dateformat an.

**design**: das Design der Uhr. **analog** steht für eine analoge Uhr, **digital** zeigt eine schöne Digitaluhr, **berlin** zeigt die Berliner Uhr (<https://en.wikipedia.org/wiki/Mengenlehreuhr>). Es gibt eine weitere Berliner Uhr namens **berlin2**, die im SVG-Format erstellt wurde. Diese ist besser für kleinere Auflösungen.

**color**: die Farbe der Segmente der Digitaluhr

Beispiel 1: einfache Textuhr

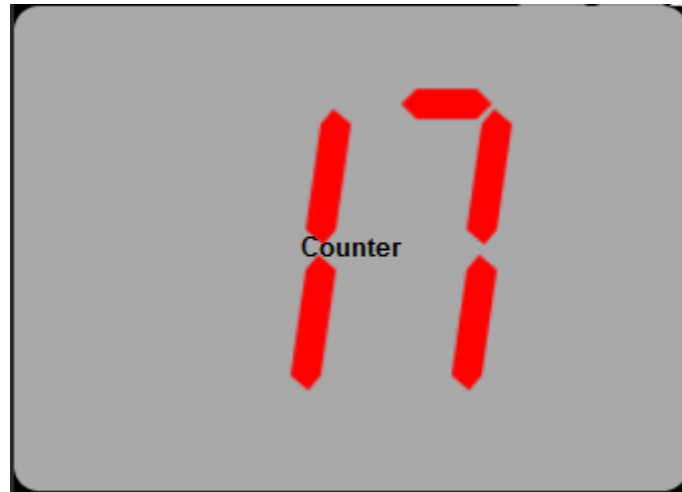
Parameter:

```
format: "15:04:05 02 Jan 06"
```

Beispiel 2: zeigt eine schöne Digitaluhr mit roten 7-Segment-Ziffern

```
format: "02-01"
design: digital
showseconds: true
color: "#ff0000"
```

## Counter



Ein einfacher Zähler mit bleibendem Wert.

Typ: ZÄHLER

Parameter:

**persist**: wahr oder falsch, wenn wahr, bleibt der Zähler zwischen Dienstneustarts bestehen

**oldschool**: Rendering eines Old-School-Zählers mit 7-Segment-Ziffern

**color**: die Farbe der Segmente

## Days remain



Zeigt die verbleibenden Tage bis zu einem Enddatum an.

type: DAYSREMAIN

Parameter:

**date**: Enddatum im Format "yyyy-mm-dd"

**formatTitle**: die Titelnachricht für die Antwort, Standardwerte %d

**formatText**: die Textnachricht für die Antwort, Standardwerte %d

**finished**: die Nachricht am Ende der Tage bleibt, Standard: finished

## Delay

Verzögert die Ausführung. Hiermit kann man 2 Kommandos, die in einer Aktion hintereinander kommen sollen, um eine gewisse Zeit verzögern.

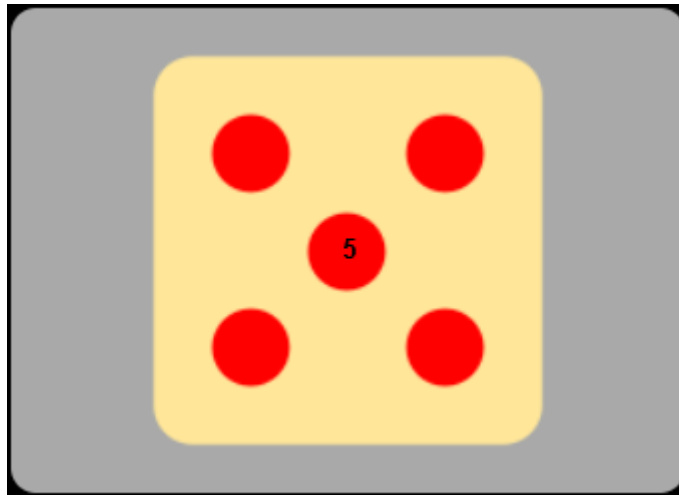
type: DELAY

Parameter:

**time**: die Verzögerung in Sekunden



## Dice



Eine einfache Würfelimplementierung mit schöner Benutzeroberfläche und verschiedenen Werten.

`type`: DICE

`Parameter`:

`sides`: die Anzahl der Seiten des Würfels. Für Seiten  $\leq 9$  ist eine Benutzeroberfläche implementiert.

## Execute

`type`: EXECUTE

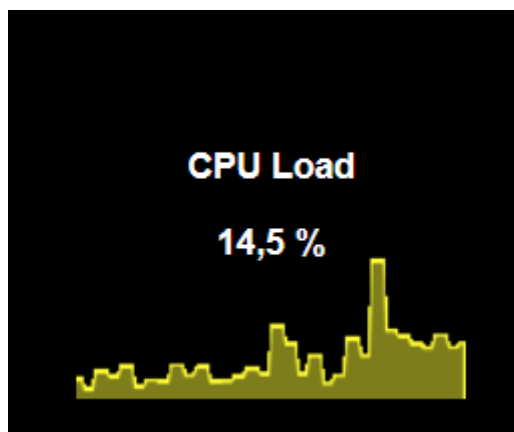
`Parameter`:

`command`: die ausführbare Datei oder das auszuführende Shell-Skript, mit oder ohne Pfad

`args`: Liste der String-Argumente für diese ausführbare Datei

`waitonClose`: Wartet mit der weiteren Ausführung, bis die ausführbare Datei geschlossen oder das Skript beendet ist.

## Hardware monitor



Dieser Befehl stellt eine Verbindung mit der Openhardware-Monitoranwendung unter Windows her. Damit können Sie Werte verschiedener Sensoren Ihres Computers erhalten. Um den Webserver der openhardwaremonitor App nutzen zu können, müssen Sie in den Einstellungen die Open Hardware Monitor Integration aktivieren. Die `URL` ist die URL zum App-Webserver, der mit `data.json` hinzugefügt wurde. `updateperiod` ist die Aktualisierungszeit in Sekunden.

Wenn Sie dies konfiguriert haben, wertet der Dienst beim Start die Verbindung und alle möglichen Sensornamen aus. Diese Namensliste sehen Sie im Log und im Konfigurationsverzeichnis gibt es eine Datei `sensorlist.txt` mit allen Sensornamen. Der Sensorname beginnt mit der Kategorie, wie CPU, GPU oder Speicher, gefolgt von der Hardwarekomponente. Danach folgt der Sensortyp wie Clocks, Temperatures oder Load, gefolgt vom Sensornamen.

`type`: `HARDWAREMONITOR`

Parameter:

`sensor`: der Sensorname.

`format`: der Formatstring für die textuelle Darstellung

`display`: Text, Grafik, Text zeigt nur die Textdarstellung, Grafik zeigt beides

`ymin`: der Wert für den Boden des Graphen

`ymax`: der Wert für den unteren Rand des Graphen

`color`: Farbe des Graphen

## Homematic

Zur Einbindung eines homematic Systems in den ReCoS gibt es 3 verschiedene Befehle.

Zuerst müssen Sie Ihre homematic in der Servicekonfiguration konfigurieren. (Siehe Homematic installieren)

Danach können Sie die 3 Befehle verwenden:

Das Starten eines Programms (**program**) ist sehr einfach: Verwenden Sie einfach den Namen als Namen für das Programm.

`name` ist der Name des hm-Programms.

Ein **Switch** dient zur Verwendung der Taste als Schalter. Einfach an/aus. Für beide Zustände können Sie ein anderes Symbol hinzufügen. Das Textfeld der Schaltfläche zeigt immer den aktuellen Zustand an. (Auch wenn Sie das Gerät über andere Software-/Hardwarelösungen wechseln)

`name` ist der Name des "Geräts: Kanal".

`officon`: das Symbol wird angezeigt, wenn der Schalter ausgeschaltet ist

`onicon`: Das Symbol wird angezeigt, wenn der Schalter eingeschaltet ist.

Für **dimmer** und **shutter** können Sie den dritten Befehl verwenden.

`name` ist das Gerät/der Kanal.

`action`: kann die verschiedenen Aktionen haben:

`set value`: Wert direkt einstellen (in Prozent)

`up`: Erhöht den Wert in Prozentschritten. Die Schrittweite ist über `value` konfigurierbar.

`down`: Verringern des Wertes in Prozentschritten. Die Schrittweite ist über `value`

konfigurierbar.

`value`: der Wert oder die Schrittweite.

Für alle Befehle ist es erforderlich, dass die Namen der Geräte/Kanäle eindeutig sind.

## Keys

Senden von Tastaturbefehlen an die aktive Anwendung. Dieser Befehl emuliert eine Tastatureingabe, indem er Tastenanschläge einer Tastatur an die aktive Anwendung sendet. Sie können verschiedene Tastaturlayouts verwenden und es gibt einige Makros, die Sondertasten definieren.

`type`: KEYS

`keyLayout`: Definiert das Layout der Tastatur, die zum Senden der Daten verwendet wird. en für Englisch (us) "qwerty" und de für ein deutsches "qwertz"-Tastaturlayout. Standard ist "de"

`keyStrokes`: sind die Zeichenfolgen mit den Tasten, die zum Senden verwendet werden. Für Sondertasten sind spezielle Makros definiert. Jedes Makro beginnt mit einem "{" und endet dann mit "}". Wenn Sie das "{" als Zeichen senden möchten, verdoppeln Sie dies einfach. ("{" -> "{{").

Ein weiteres spezialisiertes Zeichen ist das Zeichen "~". Dies führt zu einer Verzögerung von 1 Sekunde zwischen den Eingaben. Um das Zeichen "~" zu erhalten, verdoppeln Sie es einfach.

Folgende Makros sind definiert:

Macro	Keyboard key
backspace, bs, bksp	backspace
break	break
capslock	caps lock
del, delete	delete
down	arrow down
end	end
enter	enter
esc	esc
help	help
home	home
ins, insert	insert
left	arrow left
num	num lock
pgdn	page down
pgup	page up
prtsc	print screen
right	arrow right
scrolllock	scroll lock
tab	tab
up	arrow up
f1 .. f12	function key 1 ... 12

## Mediacontrol

die Kontrolle über Ihren System-Media-Player übernehmen. Mit diesen Befehlen können Sie den Mediaplayer des Systems steuern. Unter Windows kann mit diesen Befehlen normalerweise das aktuell gestartete Medienprogramm wie Windows Media Player, Spotify oder auch vlc gesteuert werden.

Die verfügbaren Befehle sind Start, Stop, Next und Previous.

**type** : MEDIAPLAY

Parameter:

**command** : Dies ist der Befehl, den Sie abfeuern möchten. **start**, startet den Mediaplayer oder pausiert ihn, wenn er bereits gestartet ist. **stop** um den Player zu stoppen und **next** und **previg** um zum nächsten/vorherigen Teil zu gehen.

## OBS - Open Broadcaster Software

Dies ist eine Reihe von Befehlen, um Ihre obs-Installation zu steuern.

### OBS Start/Stop

Start/Stop Aufnahme oder Streaming

type: OBSSTARTSTOP

Parameter:

mode: recording oder streaming, the mode to start/stop

### OBS Profile

ändert das aktuelle Profil von OBS

type: OBSPROFILE

Parameter:

profile: der Name des zu aktivierenden Profiles

### OBS Scene Collection

ändert die aktuelle Szenenkollektion

type: OBSSCENECOLLECTION

Parameter:

scenecollection: der Name Szenenkollektion

### OBS Scene

ändert die aktuelle Szene mit verschiedenen Kommandos.

type: OBSSCENE

Parameter:

scenecommand: das Szenenkommando, möglich sind:

- next: wechselt zur nächsten Szene in der Liste, am Ende wechselt es wieder zur 1. Szene
- previous: wechselt zur vorherigen Szene in der Liste, am Anfang wechselt es wieder zur letzten Szene
- first, last: wechselt direkt zur 1. oder letzten Szene in der Liste
- switch: wechselt zur der Szene mit dem angegebenen Namen

## Page

Wechselt zu einer anderen Seite im ReCoS UI.

type: PAGE

Parameter:

page: Name der Seite zu der gewechselt werden soll.

## Philips Hue

2 Commandos for controlling Philips hue lights and groups.

## Philips Hue Lights

type: PHUELIGHTS

Parameter:

**name**: the name of the hue light, group, room or zone. If a light is accessed the keyword "Light:" will be in front of the parameter, otherwise the keyword "Group:"

**brightness**: the brightness of the light (1..254, 0 for unused)

**saturation**: the saturation of the light (1..254, 0 for unused)

**hue**: the hue of the light, this is a color value ranging from 1..65535, 0 for unused

**colortemp**: the color temperature of the light, this is a value ranging from 2000..6500, 0 for unused

**color**: the color of the light

## Philips Hue Scenes

type: PHUESCENES

Parameter:

**name**: the name of the hue group, room or zone.

**scene**: the scene to apply to

**brightness**: the brightness of the group (1..254, 0 for unused)

**ATTENTION:** As names of the lights, groups, scenes ... are used here to identify the object, please be sure that in the definition of those the names are unique. Otherwise ReCoS maybe control only one of those.

## Play audio



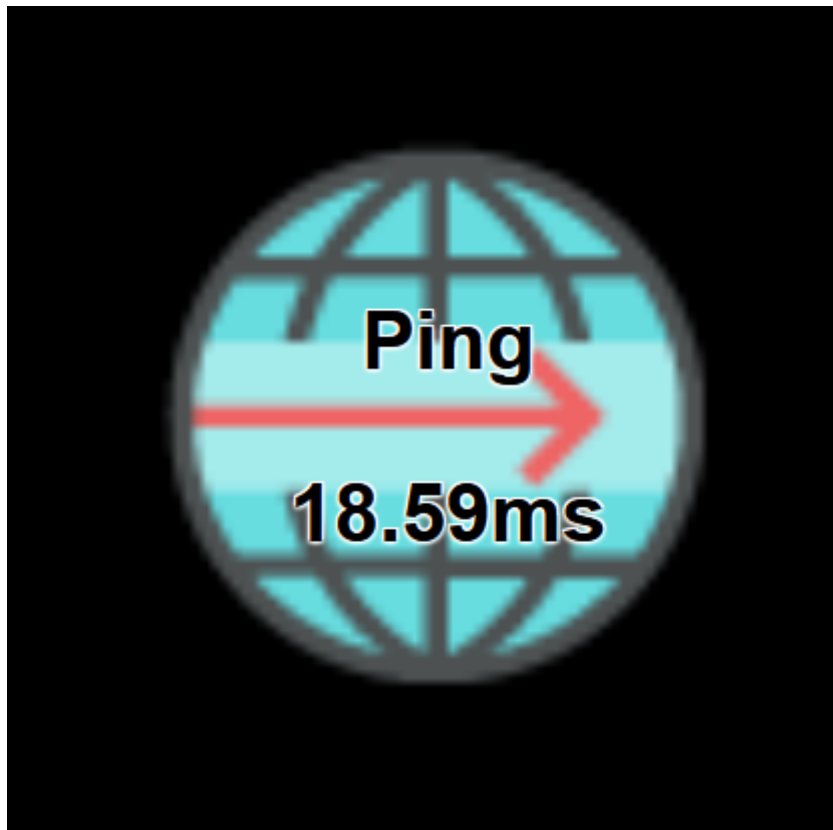
Playing an audio file. Available formats are: wav, mp3, ogg and flac. With a bunch full of this commands you can create your own soundboard.

type: PLAYAUDIO

Parameter:

**file**: the name and path to the audio file, which should be played.

## Ping



Shows ping times to a server.

Here you can test your connection to a server. Ping will test the connection to a server, You can use an IP Address or a server name (without any http...) If you put this command to an display action you can set a period so that the command will automatically start every #period seconds a test. The result is the actual ping time in ms.

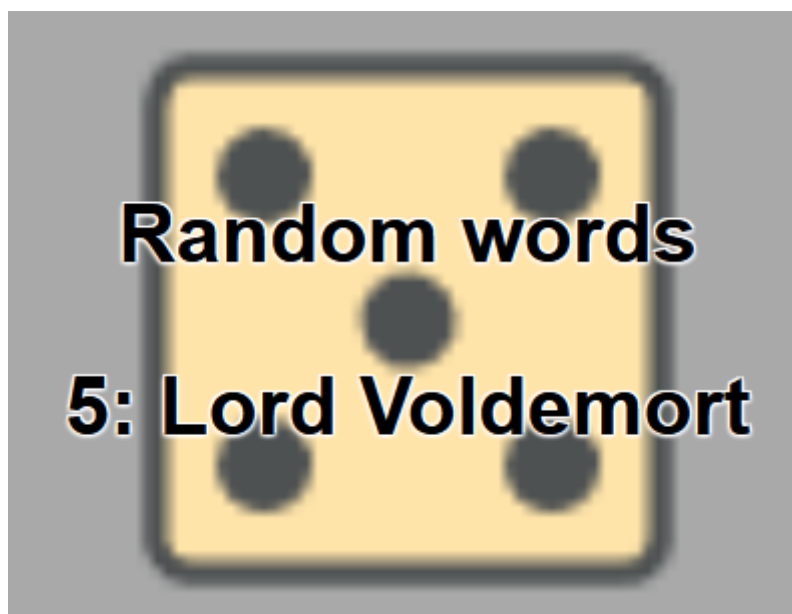
type: `PING`

Parameter:

`name`: the URL to ping to. You can use the Internet address name (without the protocol) or a simple IP address

`period`: the period in seconds when the command should automatically be executed.

## Random words



Choose a random word/phrase out of a list

`type`: RNDWORDS

Parameter:

`words`: The list of words/phrases to choose one randomly from

## Screen shot

taking a screen shot.

With this command, you can take a screen shot.

`type`: SCREENSHOT

Parameter:

`saveto`: the folder, where the screen shot will be saved. File name format is

`screen_<#number>_<display>.png`

`display`: optional, the number of the display, if you want to store screen shot of every display please use -1. Getting the right display, simply do a screen shot with `display = -1`. Then look at the screen shots and look in the name at the last number of the right image. That is your display.

## Send message

`type`: SENDMESSAGE

Parameter:

`receiver`: the receiver of this message. `client` is for all clients having this particular action active, `service` is for the service itself. (Or more the user sitting on the service related desktop)

`message`: the message to send

## Show text

Showtext will show a text on the button, and the icon of this command, if set, will be displayed as the title, too.

`type`: SHOWTEXT

Parameter:

`text`: the text to show in the text field of the button.

## Start Browser

`type`: BROWSE

Parameter:

`url`: the URL to show in the system browser. On Windows if you choose a normal file system folder, it will automatically start the explorer on this path.

Example 1

start a new browser windows with a url:

```
url: https://www.wk-music.de
```

Example 2

on windows: start a new explorer windows with a filepath



```
url: c:\windows
```

## Stopwatch

A simple textual stopwatch.

**type:** STOPWATCH

**Parameter:**

**format:** the format of the time. The `%` character signifies that the next character is a modifier that specifies a particular duration unit. The following is the full list of modifiers supported by `go-durationfmt`:

- `%y` - # of years
- `%W` - # of weeks
- `%d` - # of days
- `%h` - # of hours
- `%m` - # of minutes
- `%s` - # of seconds
- `%%` - print a percent sign

You can place a `0` before the `h`, `m`, and `s` modifiers to zeropad those values to two digits. Zeropadding is undefined for the other modifiers.

## Timer

Starting a timer with a response every second. You can define the format of the timer message and the message on finish.

**type:** TIMER

**Parameter:**

**time:** time to delay in Seconds

**format:** the message for the response, defaults `%d` seconds

**finished:** the message at the end of the timer, defaults: finished

## WindowCtrl

Controlling Application Main Window.

With this command, you can control the main window of an application.

**type:** WINDOWCTRL

**Parameter:**

**caption:** the caption of the application window

**command:** the command to execute on this window. Possible values are:

• **minimize:** for minimizing the application window

• **activate:** for activating the application window again. (restore it if minimized and active/bring it to front)

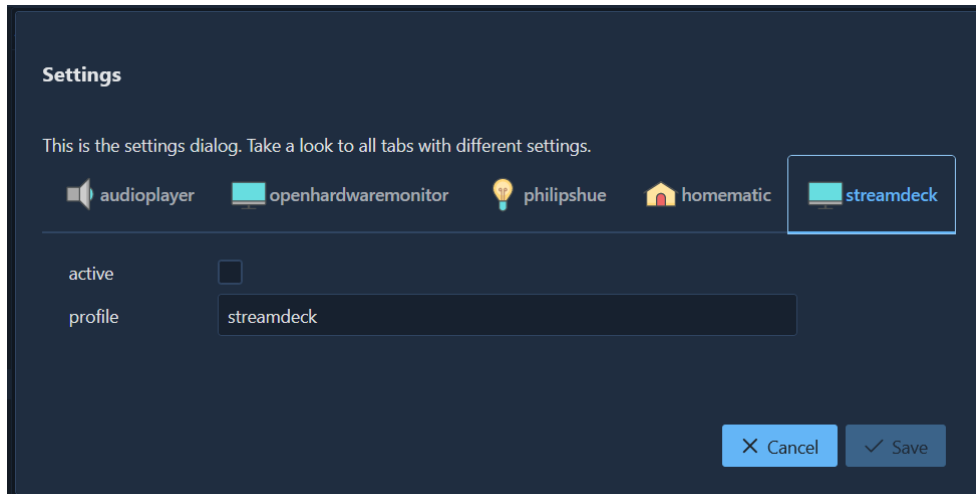
• **move x y:** moving the window to the new position x,y

# Installing 3'rd Party products

For other 3rd party products there is an so called integration for accessing the different parts. Here you can find some remarks on these integration.

## Elagto Stream Deck (c)

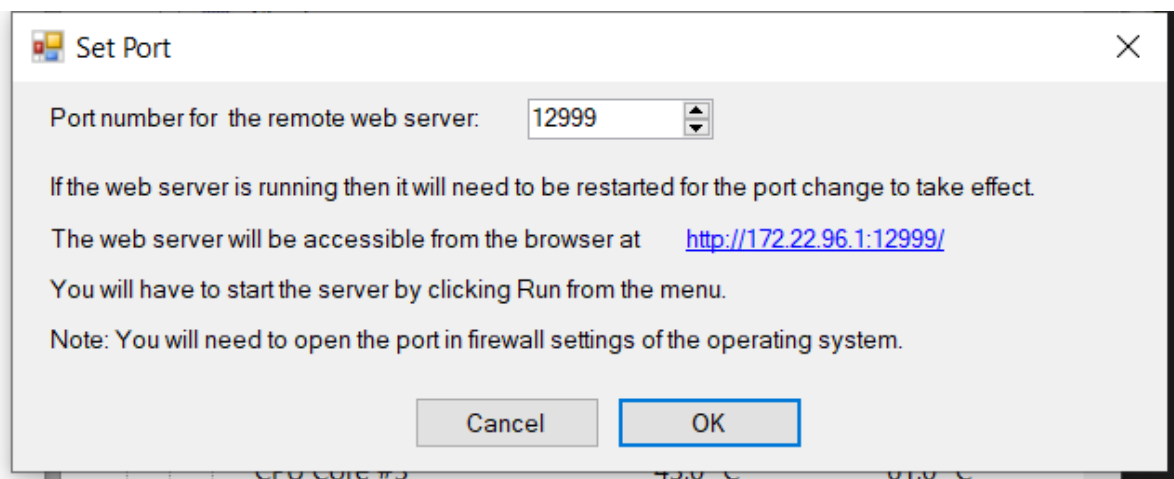
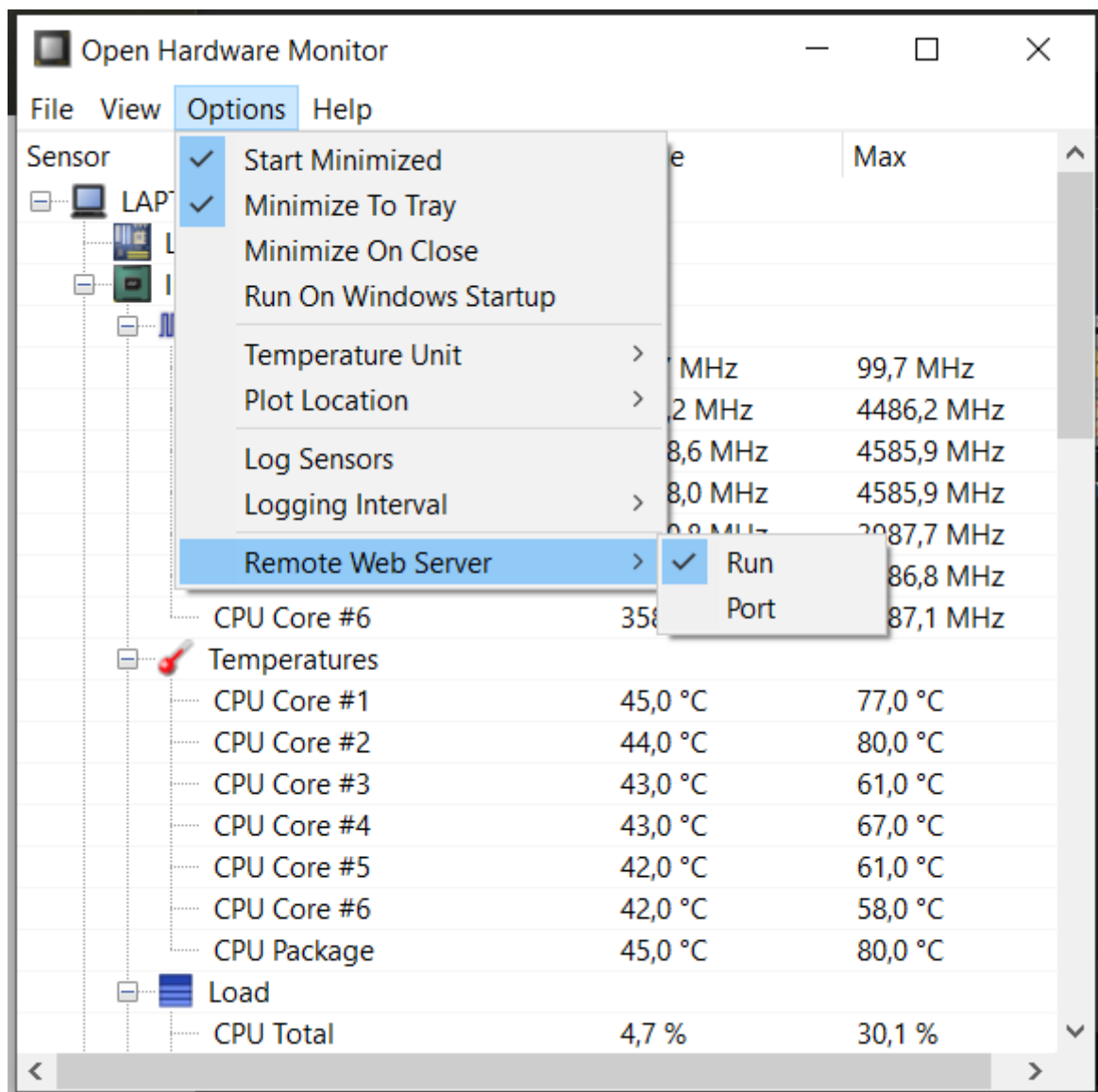
The stream deck integration instrumented an Elgato stream deck for the ReCoS system. Since only one application can access the hardware, it is necessary that you deactivate and close the original stream deck application. You can then simply activate the stream deck integration in the section settings in the admin client.



The second parameter is optional. You can add here the profile, the stream deck will present. But there are defaults for it. For a normal stream deck (15 Button version, no matter whether revision 1 or 2) the default profile is called `streamdeck`. A profile with the name `streamdeck_mini` is expected for the Stream deck mini. The `streamdeck_xl` profile is expected for the XL. The following applies to all three, if the profile is not found and no profile is specified, the `default` profile is used. So that the profiles in the surface correspond to the display in the stream deck, you should create the rows and columns accordingly. For the normal stream deck the configuration is 3x5 (rows / cols) for the Mini 2x3 and for the XL 4x8. Other profile configurations also work, but it is possible that not all buttons are displayed on the stream deck or that they remain empty.

## Installation of OpenHardwareMonitor

For hardware sensor reading ReCoS uses the OpenHardwareMonitor Software. (<https://openhardwaremonitor.org/>) To use this, simply install the software. After installation, go to Option/Remote Web Server/Port.

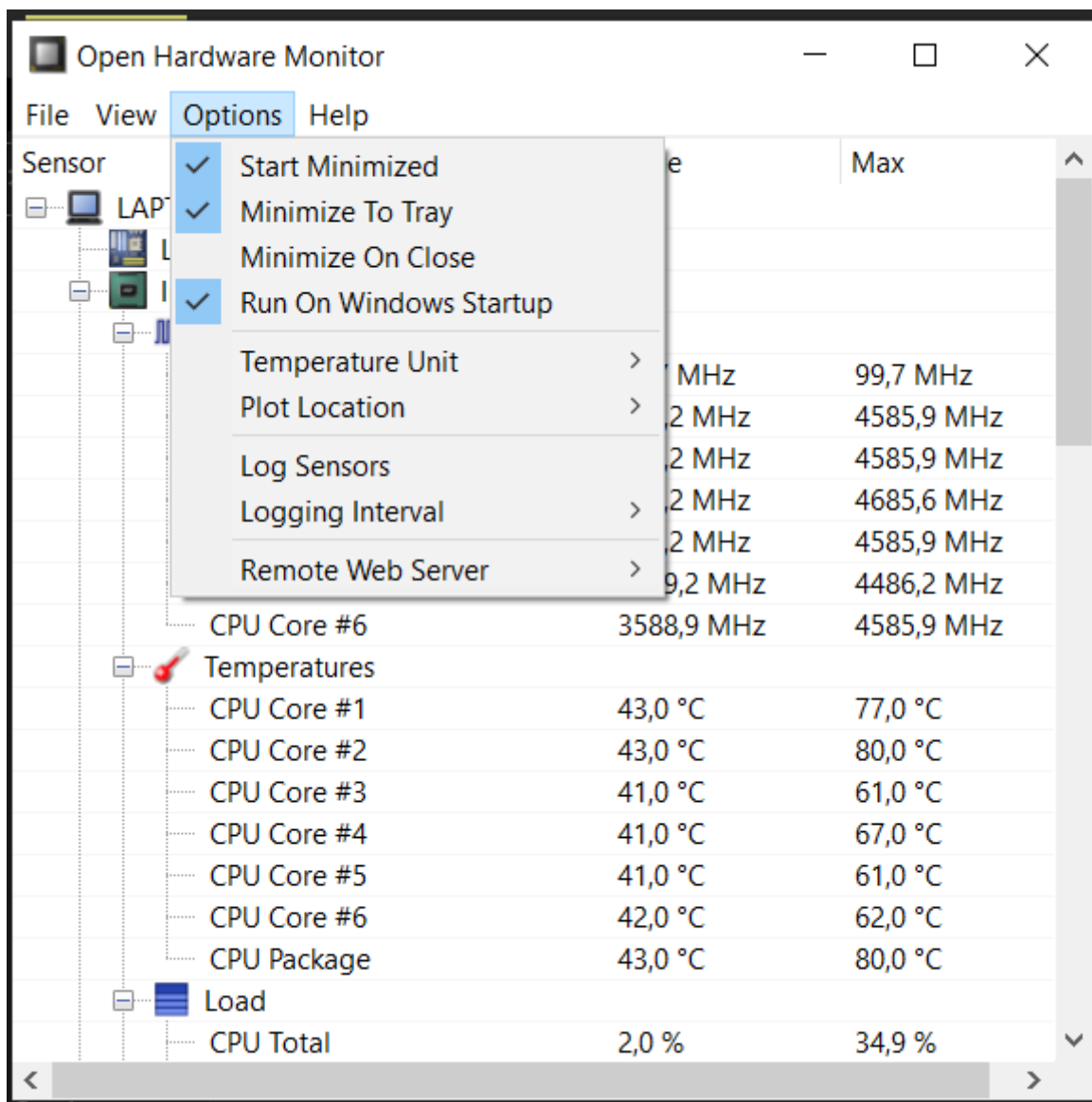


As Port number enter 12999 (which is the default for Combination of ReCoS and OHM)

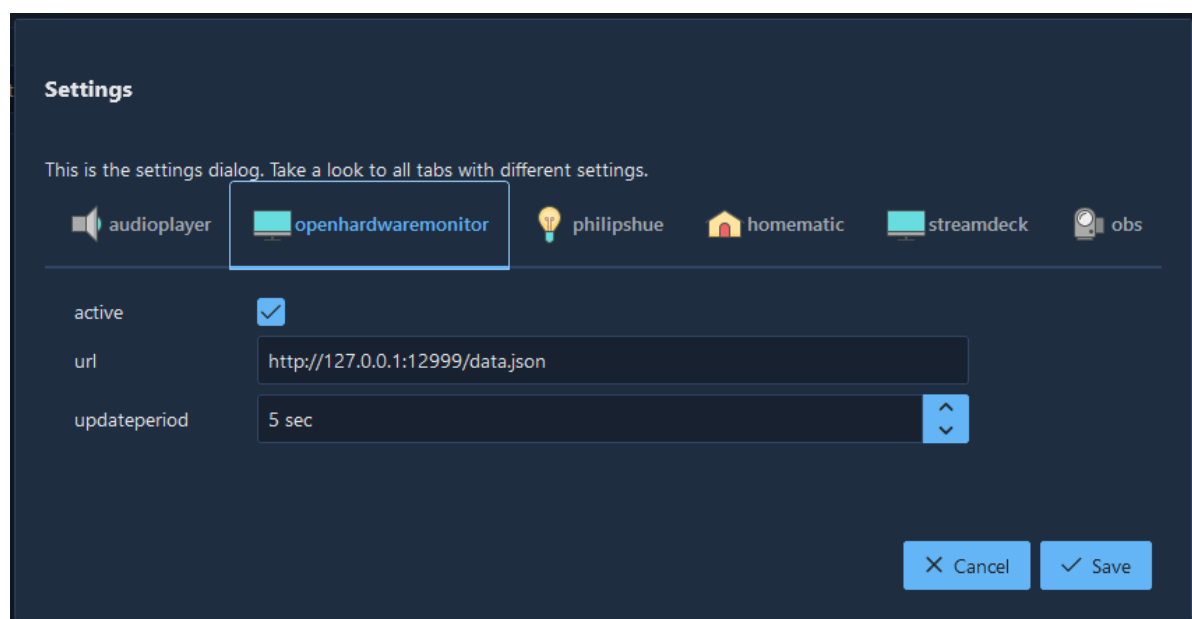
After that simply activate the OHM Web server via Option/Remote Web Server/Run.

The OHM should be available after restart of windows, so please tick the following options on.

Now everything of the OHM side is ready. The ReCoS service will now automatically connect to the OHM and get all Sensors.



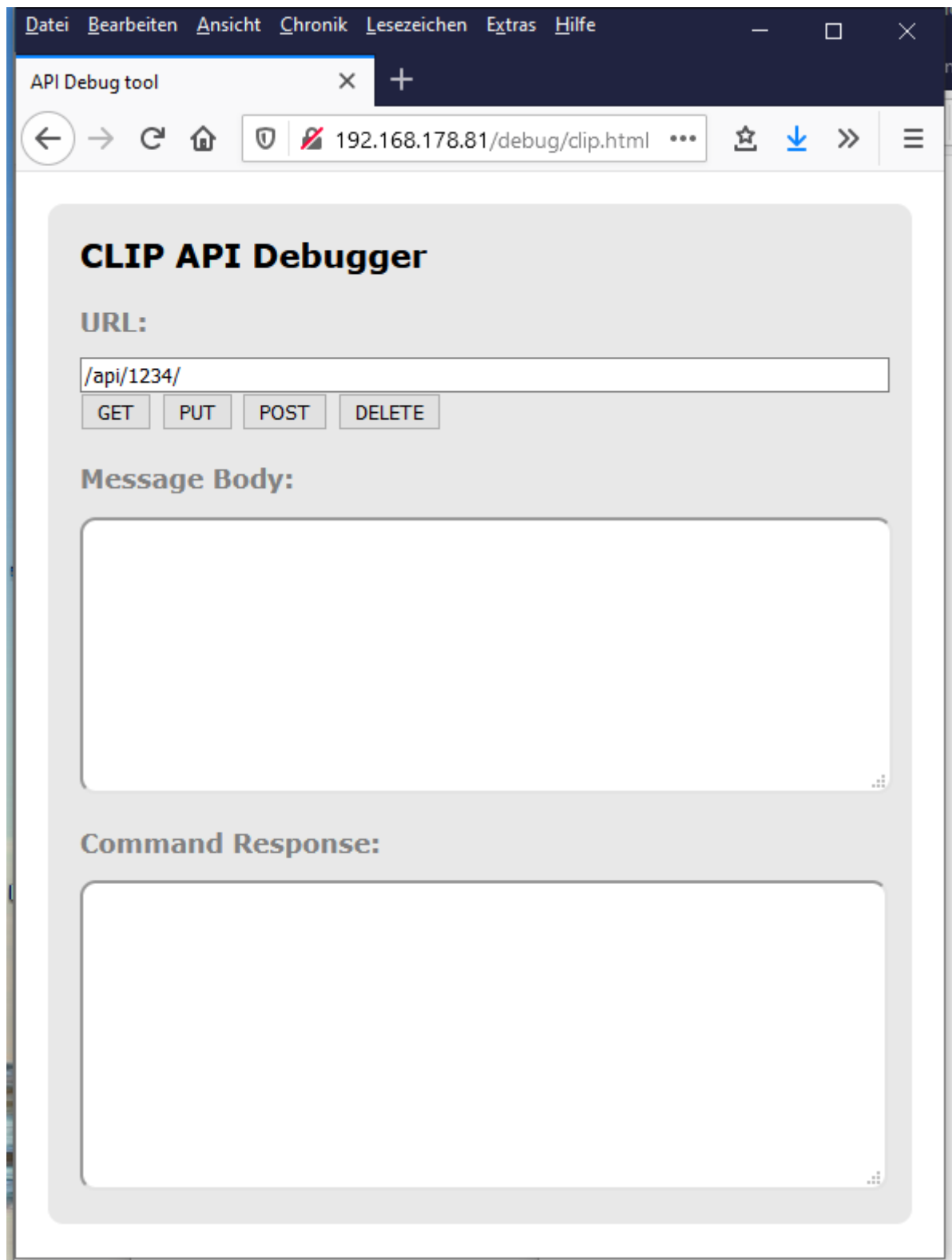
If you have already installed the OHM (maybe for another App) and you can't reconfigure the port option, you can simply change the port for ReCoS in the admin client. Same for the url, when you want to get sensors from another Computer.



## Installation Philips Hue

First of all, a new user must be created for the ReCoS installation. To do this, please start your browser and go to the clip page of your Hue Bridge.

<http://192.168.178.81/debug/clip.html>

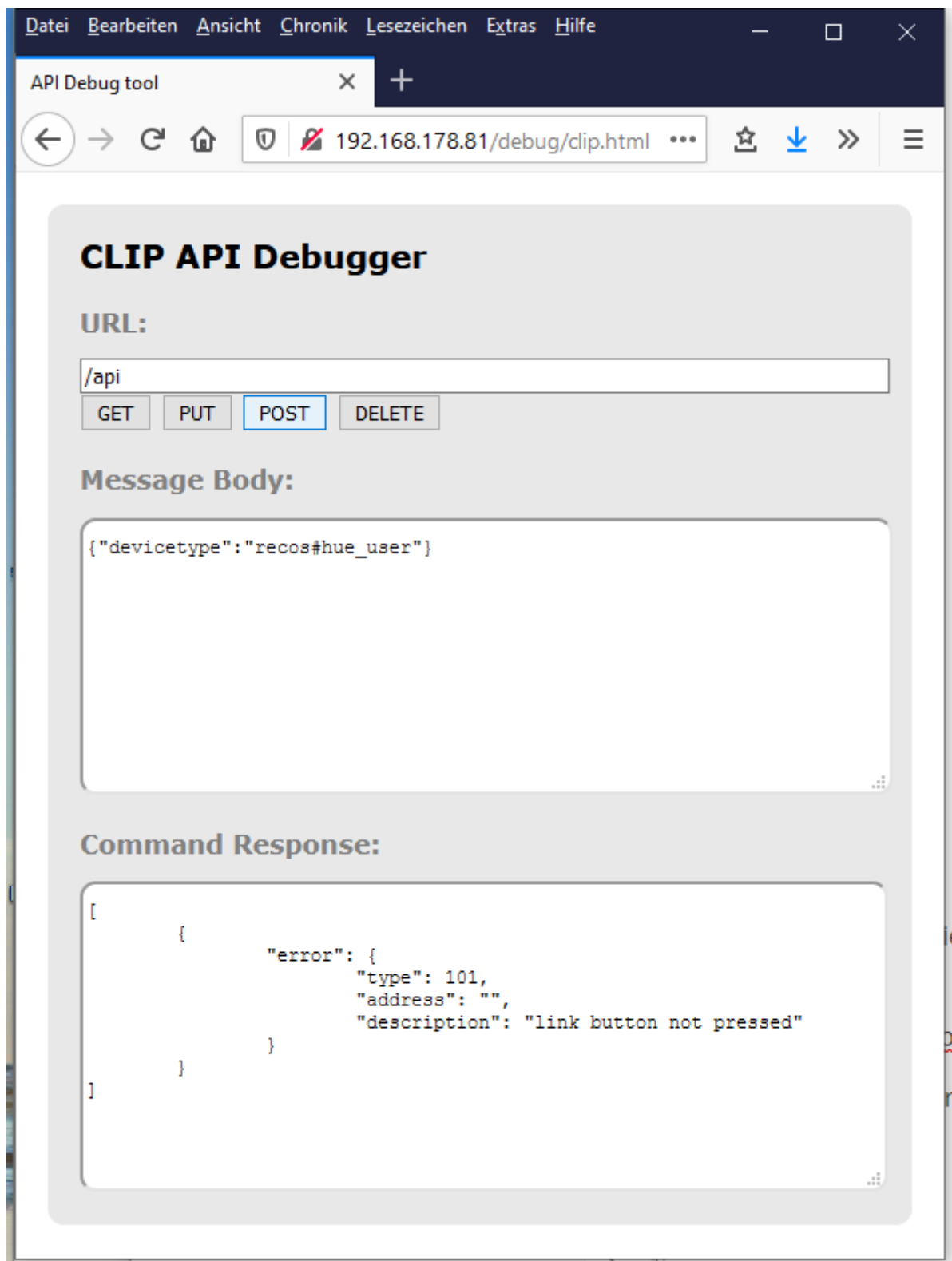


Now please post the following to the bridge.

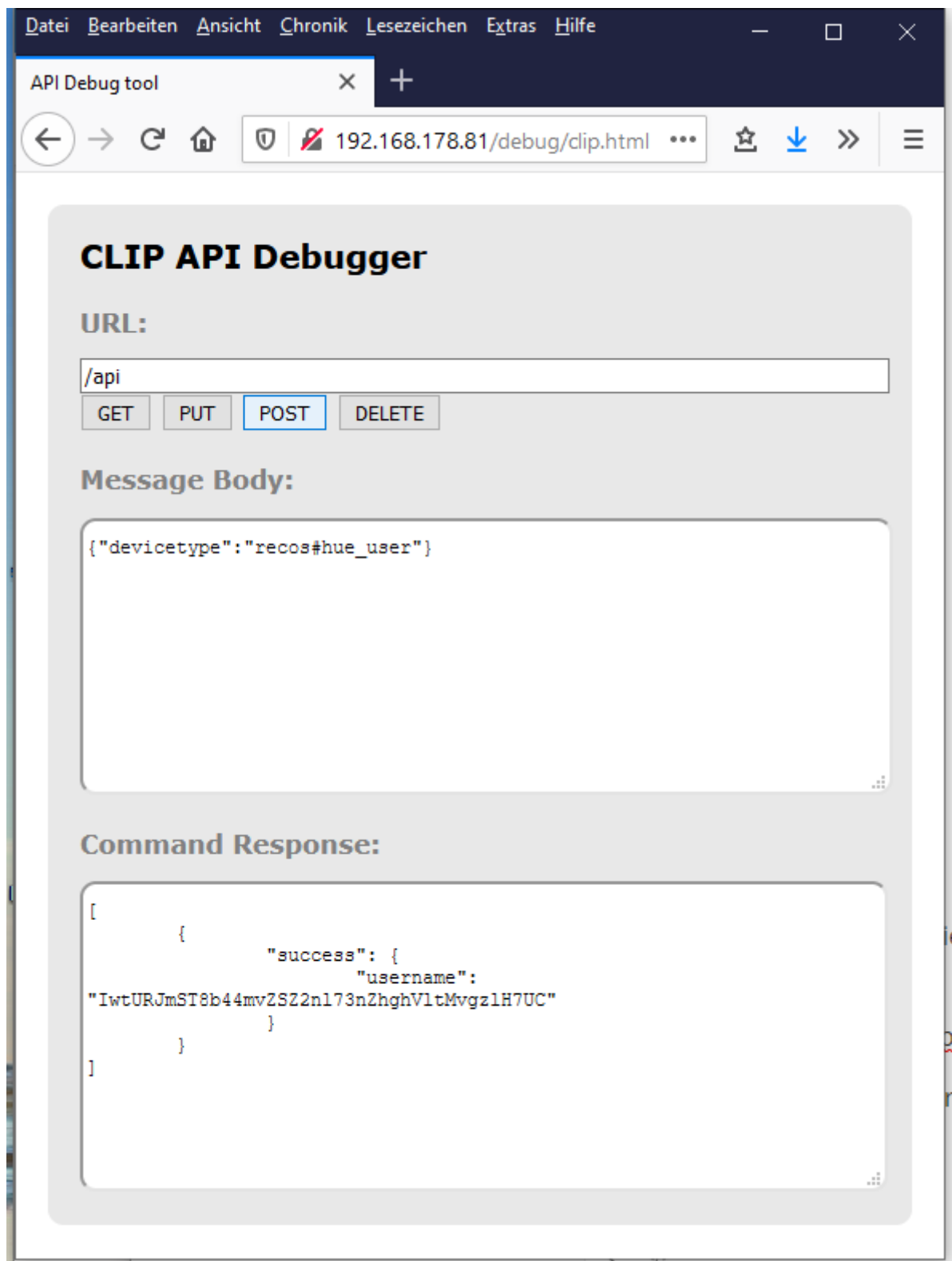
URL: /api

Message Body: {"devicetype":"recos#hue\_user"}

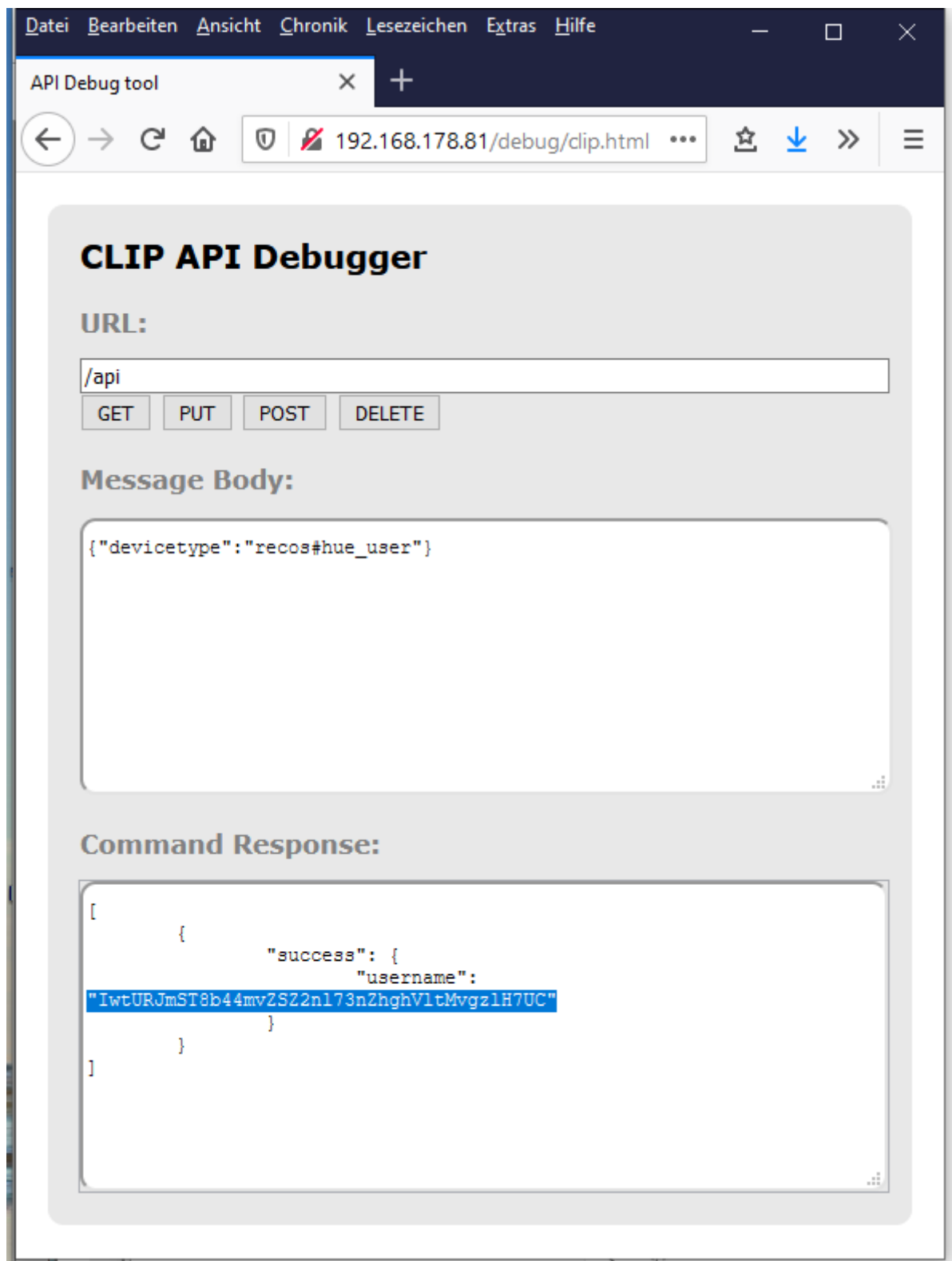
With the first attempt an error message appears.



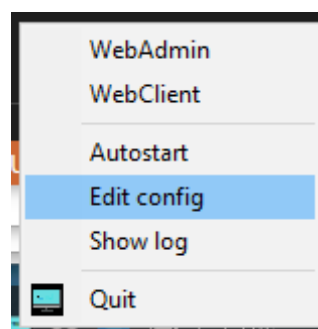
Please press the big button on the Philips Hue Bridge and post the same thing again.



Then copy the username from the answer:



Now you have to change the ReCoS configuration. To do this, please start the admin client via the context menu.





**Settings**

This is the settings dialog. Take a look to all tabs with different settings.

audioplayer openhardwaremonitor **philipshue** homematic streamdeck obs

active ☒

username <the bridge generated username here>

device recos#hue\_user

ipaddress 127.0.0.1

updateperiod 5 sec

Cancel Save

Please enter the generated user name here as the `username` and change the `ipaddress` accordingly.  
`updateperiod` is the time span in seconds in which ReCoS polls the bridge for changes.

## Homematic

For using the Homematic system for commands, you simply have to do 2 Things:  
 First you have to add the xmlapi addon to your homematic.

<https://github.com/homematic-community/XML-API>

Second: add the url of your homematic system in the system config (via admin client/settings) and set the active state to true. After a redstart of the ReCoS you can use the homematic commands.

**Settings**

This is the settings dialog. Take a look to all tabs with different settings.

audioplayer openhardwaremonitor philipshue **homematic** streamdeck obs

active ☐

url http://192.168.172.10

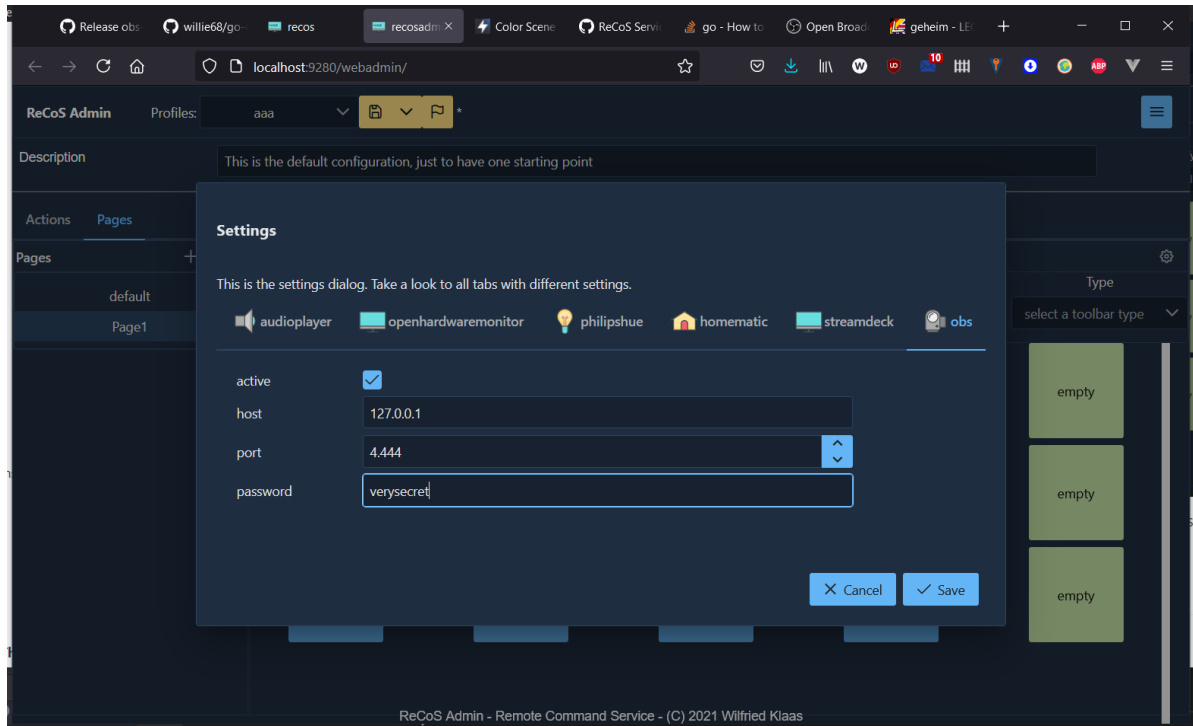
updateperiod 5 sec

Cancel Save

## OBS Studio - Open Broadcaster Software

With ReCoS you can control some parts of your OBS Software. This integration supports the OBS Studio. First you have to install the obs-websocket plugin. At the moment only the version 4.9.0 is supported. (<https://github.com/Palakis/obs-websocket/releases/tag/4.9.0>)

To activate the integration simply go to the settings page.



The host is the pc where the obs is installed. Normally this is the same pc, so 127.0.0.1 should work. The default port is 4444. (If you don't change it in the settings of the plugin) On the password field use the password you setup in the plugin settings (or leave it empty, if no password was been set.)

Thats all for setting up the integration.

## For the nerds

### Profile configuration

Every profile has it's own configuration file. The file is usually located in the Profiles folder. This file is written in yaml and has the following sections:

In the root you will find the following parameters

- `name`: The name of the profile
- `description`: a user readable description
- `pages`: This are the different pages for showing up the actions
- `actions`: a list of different actions. An action can appear on different pages. But as it's the same action, the status/result will be the same.

This is an example

```
name: default
description: This is the default configuration, just to have one starting point
pages:
  - name: Default
    description: Some default command for every operating system
    columns: 3
    rows: 2
    toolbar: ""
    cells:
      - Hello_World
      - clocks
```

```

- timer
- analogclock
- cpuload
- stopwatch
- name: clocks
  description: clocks only
  columns: 2
  rows: 2
  toolbar: hide
  cells:
    - Back
    - none
    - clock
    - analogclock
actions:
- type: SINGLE
  name: Hello_world
  title: Hello world
  icon: chat.png
  description: Execute notepad and wirte Hello ReCoS to it.
  fontsize: 0
  fontcolor: ""
  outlined: false
  runone: true
  commands:
    - id: EXECUTE_0
      type: EXECUTE
      name: execute
      description: ""
      icon: ""
      title: ""
      parameters:
        args: []
        command: notepad.exe
        waitonClose: false
    - id: DELAY_1
      type: DELAY
      name: delay
      description: ""
      icon: ""
      title: delay
      parameters:
        time: 3
    - id: KEYS_2
      type: KEYS
      name: typeit
      description: ""
      icon: ""
      title: typeit
      parameters:
        keys: Hello ReCoS
        layout: de
  actions: []
- type: SINGLE
  name: clocks
  title: clocks
  icon: clock.png
  description: Goto clocks page

```

```

    fontsize: 0
    fontcolor: ""
    outlined: false
    runone: true
    commands:
      - id: PAGE_3
        type: PAGE
        name: page
        description: ""
        icon: ""
        title: ""
        parameters:
          page: clocks
        actions: []
- type: SINGLE
  name: timer
  title: Timer 10 sec
  icon: alarm_bell.png
  description: timer counting down 10 sec
  fontsize: 0
  fontcolor: ""
  outlined: false
  runone: true
  commands:
    - id: TIMER_4
      type: TIMER
      name: timer
      description: ""
      icon: ""
      title: ""
      parameters:
        finished: Fertig
        format: noch %ds
        time: 10
      actions: []
- type: DISPLAY
  name: analogclock
  title: Analogclock
  icon: ""
  description: ""
  fontsize: 0
  fontcolor: white
  outlined: false
  runone: true
  commands:
    - id: CLOCK_5
      type: CLOCK
      name: clock
      description: ""
      icon: ""
      title: ""
      parameters:
        analog: true
        format: "15:04:05\r\n02 Jan 06"
      actions: []
- type: SINGLE
  name: screenshot
  title: Screenshot

```

```

icon: monitor.png
description: doing a screenshot
fontsize: 0
fontcolor: ""
outlined: false
runone: true
commands:
  - id: SCREENSHOT_6
    type: SCREENSHOT
    name: screenshot
    description: ""
    icon: ""
    title: ""
    parameters:
      display: 1
      saveto: e:/temp/screenshot
  actions: []
- type: DISPLAY
  name: cpuload
  title: CPU Load
  icon: company.png
  description: cpu usage
  fontsize: 0
  fontcolor: white
  outlined: false
  runone: true
  commands:
    - id: HARDWAREMONITOR_7
      type: HARDWAREMONITOR
      name: cpu
      description: ""
      icon: ""
      title: ""
      parameters:
        color: '#ffff00'
        display: both
        format: '%0.1f %'
        sensor: CPU/Intel Core i7-6820HQ/Load/CPU Total
      actions: []
- type: SINGLE
  name: stopwatch
  title: Stopwatch
  icon: timer.png
  description: simple stop watch
  fontsize: 0
  fontcolor: black
  outlined: true
  runone: true
  commands:
    - id: STOPWATCH_8
      type: STOPWATCH
      name: stp1
      description: ""
      icon: ""
      title: ""
      parameters:
        format: 'Mom: %0m:%0s'
      actions: []

```

```
- type: SINGLE
  name: clock
  title: text clock
  icon: clock.png
  description: ""
  fontsize: 0
  fontcolor: ""
  outlined: false
  runone: false
  commands:
    - id: CLOCK_9
      type: CLOCK
      name: clock
      description: ""
      icon: ""
      title: clock
      parameters: {}
  actions: []
- type: SINGLE
  name: Back
  title: ""
  icon: ""
  description: ""
  fontsize: 0
  fontcolor: ""
  outlined: false
  runone: false
  commands:
    - id: PAGE_10
      type: PAGE
      name: Back
      description: ""
      icon: ""
      title: Back
      parameters:
        page: Default
  actions: []
```