

# NEO AUTOMATION MANAGER

## How-To Dokumentation



## EINLEITUNG

Der NEO Automation Manager ist ein leistungsstarkes AIO CREATOR NEO Plugin zur Erzeugung automatisierter Abläufe über Regeln und Bedingungen. Mit ihm können alle in das AIO CREATOR NEO System integrierten Hersteller und Smart Home Systeme plattform- und technologieübergreifend automatisiert werden. Um den NEO Automation Manager nutzen zu können, benötigen Sie die AIO CREATOR NEO Version 2.0 oder höher.

Zur Erzeugung der Automatik-Programme, **den Tasks**, verfügt der NEO Automation Manager über ein grafisches DIY-Benutzer-Frontend, den **Blockeditor**, zur einfachen Realisierung von Basisbedingungen ohne Programmierkenntnisse. Darüber hinaus bietet er mit dem **Skripteditor** ein Profi-Frontend zur Erzeugung beliebig komplexer Zusammenhänge mittels JavaScript.

Zur Überwachung der Ausführung der mit dem NEO Automation Manager erzeugten Tasks dient das Programm **NEO SERVER**. Um diese Aufgabe wahrnehmen zu können, muss der NEO SERVER auf einem Rechner innerhalb des Netzwerks installiert und gestartet sein. Ist der Rechner ausgeschaltet oder das Programm NEO SERVER läuft nicht im Hintergrund, können keine Tasks ausgeführt werden. Der NEO SERVER muss sich im selben Netzwerk befinden wie die in NEO integrierten Geräte. Auf dem AIO GATEWAY V5 Plus ist NEO SERVER bereits vorinstalliert, muss aber unter Umständen erst noch aktiviert werden. Dies geht ebenfalls über den NEO Automation Manager.

## INSTALLATION DES NEO SERVER

Um den NEO SERVER verwenden zu können, muss dieser auf einem der folgenden Geräte laufen:

- AIO GATEWAY V5 Plus
- Rechner mit Windows, macOS oder Linux
- Raspberry Pi

### AIO GATEWAY V5 Plus

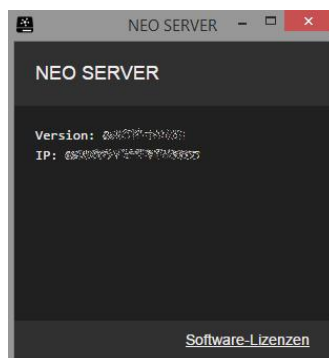
Auf dem V5 Plus ist der NEO SERVER bereits vorinstalliert, sodass Sie in diesem Fall den NEO SERVER nicht zusätzlich aufspielen müssen. Eventuell müssen Sie NEO SERVER aber noch im Automation Manager aktivieren. Falls dies nötig ist, erscheint im Automation Manager ein „Freischalten“-Button.

### Windows, macOS und Linux

Installieren Sie zunächst das Programm NEO SERVER für Ihr Betriebssystem. Die Installationsdateien können Sie unter den folgendem Link herunterladen:

<http://www.mediola.com/service#downloads?type=software&product=neoserver>

Starten Sie nun das Programm. Es erscheint folgendes Programmfenster.



Nun ist der NEO SERVER gestartet. NEO SERVER muss durchgängig weiterlaufen um seine Serverfunktion auszuführen.

### Raspberry Pi

#### 1. Installieren Sie zunächst das Programm „nodejs“ LTS version (6.9.5) auf dem Raspberry Pi

→ Gehen Sie in das „home“ Verzeichnis des aktuellen Benutzers

```
cd
```

→ Laden Sie das Programm „nodejs“ herunter und entpacken Sie es anschließend

```
wget https://nodejs.org/dist/v6.9.5/node-v6.9.5-linux-armv6l.tar.gz
```

```
tar -xvf node-v6.9.5-linux-armv6l.tar.gz
```

```
cd node-v6.9.5-linux-armv6l
```

→ Kopieren Sie es in das Verzeichnis „/usr/local“

```
sudo cp -R * /usr/local/
```

→ Um zu überprüfen, ob Node.js richtig installiert wurde und ob es die richtige Version ist, führen Sie das Kommando `node -v` aus. Als Antwort sollten Sie `v6.9.5` erhalten.

## 2. Installieren Sie nun den „NEO SERVER“ auf dem Raspberry Pi

→ Gehen Sie in das „home“ Verzeichnis des aktuellen Benutzers

```
cd
```

→ Laden Sie das Programm „NEO SERVER“ herunter und entpacken Sie es anschließend

```
wget https://s3-eu-west-1.amazonaws.com/mediola-download/neo_server/dist/NEO+SERVER-rpi.tar.gz
mkdir neo_server
cd neo_server
tar -xvf ../NEO+SERVER-rpi.tar.gz
```

→ Starten Sie jetzt den NEO SERVER.

```
sudo node automation.js
```

→ Um den so gestarteten NEO SERVER zu beenden, drücken Sie STRG+C.

## 3. Starten Sie den „NEO SERVER“ als System-Dienst

→ Zunächst müssen Sie in der Datei `nano neo_server_daemon` die „HOMEPATH“-Variable anpassen, sodass diese auf das NEO SERVER-Verzeichnis auf Ihrem Raspberry Pi verweist. Wenn Sie den Pfad zu Ihrem Benutzerverzeichnis nicht kennen, können Sie diese über den Befehl `echo $HOME` ausgeben lassen. Üblicherweise setzt sich der Pfad aus „/home/“ und Ihrem Benutzer auf dem Raspberry Pi zusammen.

→ Bearbeiten Sie nun die Datei „neo\_server\_daemon“:

```
nano neo_server_daemon
```

Navigieren Sie mit der Tastatur an die entsprechende Stelle und tragen Sie hinter „HOMEPATH=“ den korrekten Pfad ein.

```
GNU nano 2.5.3 Datei: neo_server_daemon
#!/bin/sh
### BEGIN INIT INFO
# Provides: neo_server_daemon
# Required-Start: $network $remote_fs $syslog
# Required-Stop: $remote_fs $syslog
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start daemon at boot time
# Description: Enable service provided by daemon
### END INIT INFO
HOMEPATH=/home/pi/neo_server
PID=$(pidof node automation.js)
case "$1" in
  start)
    if [ ! -z "$PID" ]; then
      echo "neo_server is already running"
    else
      cd "$HOMEPATH"
      nice -10 sudo node automation.js > /var/log/neo_server_daemon 2->&1 &
      echo "neo_server starting"
      sleep 2
      "$@" status
    fi
  ;;
  stop)
    if [ -z "$PID" ]; then
      echo "neo_server is not running"
    else

```



Speichern (STRG+O, Enter) und schließen (STRG+X) Sie die Datei anschließend.

→ Kopieren Sie die Datei nach „/etc/init.d“:

```
sudo cp neo_server_daemon /etc/init.d
```

→ Modifizieren Sie die Daemon-Skript-Berechtigung und registrieren Sie abschließend das Skript für den Autostart.

```
cd /etc/init.d
```

```
sudo chmod 0755 neo_server_daemon
```

```
sudo update-rc.d neo_server_daemon defaults
```

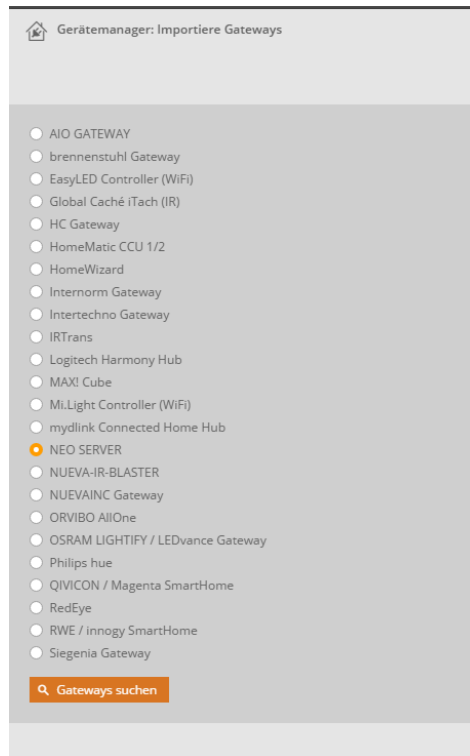
## INTEGRATION DES NEO SERVER IN DEN AIO CREATOR NEO

Haben Sie den NEO SERVER installiert und gestartet, müssen Sie ihn im Gerätemanager des AIO CREATOR NEO als neues Gateway einbinden.

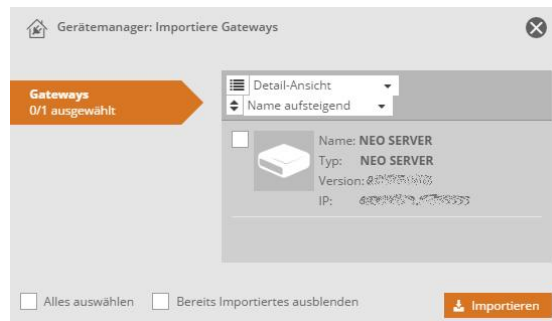
**i** Denken Sie bitte daran, Ihr laufendes System vorher zu sichern.

### NEO SERVER über Windows, macOS , Linux oder Raspberry Pi in NEO integrieren

Öffnen Sie nun den Tab „Gateways“ des NEO Gerätemanager und klicken auf „Gateways suchen“.



Im folgenden Auswahlmenü selektieren Sie „NEO SERVER“ und klicken anschließend auf „Gateways suchen“.

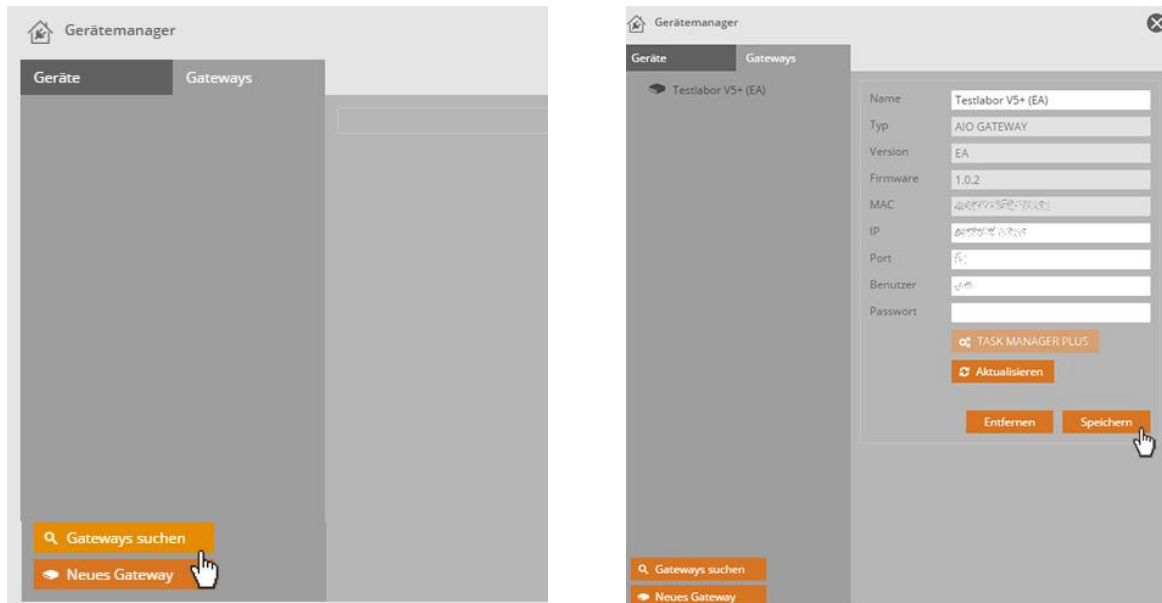


Nun wird der NEO SERVER angezeigt und kann durch Anklicken der „Importieren“-Taste als neues Gateway in den Geräte-Manager importiert werden. Anschließend wird der NEO SERVER als Gateway im Gateways-Tab aufgelistet und man kann dort einen individuellen Namen vergeben. Haben Sie diesen Vorgang abgeschlossen, können Sie den NEO SERVER im NEO Automation Manager auswählen und Tasks erstellen.

## AIO Gateway V5 Plus in NEO integrieren

Nutzen Sie den Automation Manager mit dem V5 Plus, dann integrieren Sie das V5 Plus in den AIO CREATOR NEO, indem Sie im Geräte-Manager im Tab „Gateways“ auf „Gateways suchen“ klicken. Im anschließenden Auswahlmenü wählen Sie „AIO GATEWAY“ und klicken auf „Gateways suchen“.

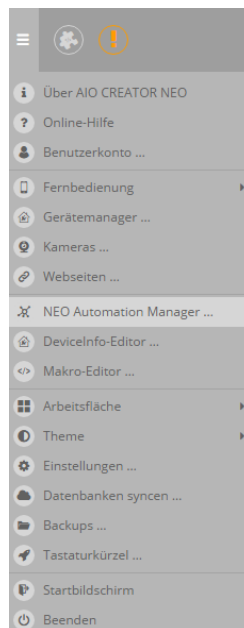
Nachdem das V5 Plus angezeigt wird, kann es über die „Importieren“-Taste in NEO integriert werden. Auch hier können Sie nach dem Import einen eigenen Namen für das V5 Plus vergeben.



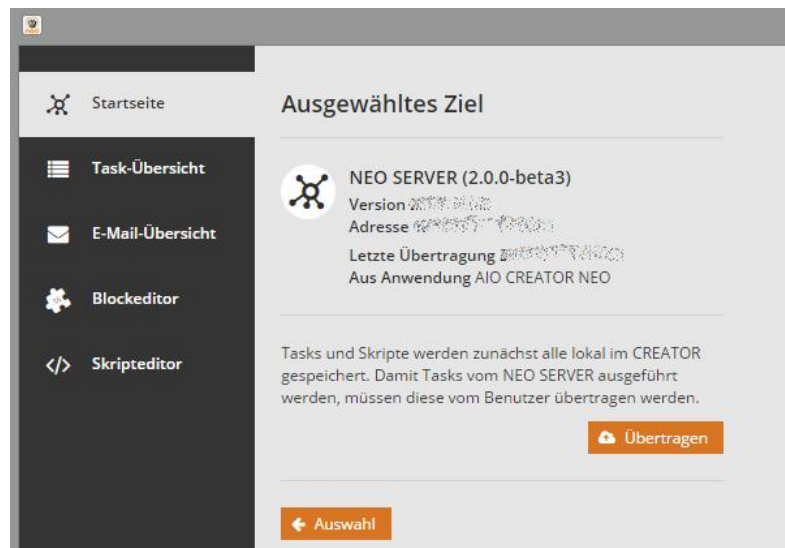
Jetzt kann das V5 Plus im NEO Automation Manager als Server ausgewählt werden.

## ERZEUGEN VON TASKS MIT DEM NEO AUTOMATION MANAGER

Der NEO Automation Manager wird im Einstellungsmenü des AIO CREATOR NEO geöffnet.



Je nachdem, ob Sie ein V5 Plus nutzen oder ob NEO SERVER bei Ihnen auf einem PC oder Raspberry Pi läuft, sehen Sie auf der Startseite den zuvor integrierten NEO SERVER bzw. das zuvor integrierte V5 Plus als ausgewähltes Ziel zur Übertragung der im NEO Automation Manager erzeugten Tasks.



**i** Im Testmodus können bis zu zwei Tasks gleichzeitig auf dem NEO SERVER aktiv sein. Um eine beliebige Anzahl Tasks freizuschalten, ist eine kostenpflichtige Plugin-Lizenz notwendig.

Grundsätzlich werden alle Tasks und Skripte, die im NEO Automation Manager erstellt oder verändert wurden, zunächst nur lokal im AIO CREATOR NEO gespeichert. Damit die Tasks ausgeführt bzw. Veränderungen eines Tasks übernommen werden können, müssen diese auf den NEO SERVER / das V5 Plus übertragen werden. Nutzen Sie zum Übertragen die „Übertragen“-Taste auf der Startseite.

Nach der Übertragung wird die Ausführung der Tasks allein vom NEO SERVER / V5 Plus überwacht und durchgeführt (der AIO CREATOR NEO wird dazu nicht benötigt).

## Erstellen von Tasks

Tasks werden grundsätzlich im Blockeditor erzeugt.

### Blockeditor

Der Blockeditor dient zur einfachen Erzeugung von automatischen Abläufen, die an Bedingungen und Regeln gekoppelt werden. Diese Abläufe werden als Tasks gespeichert. Tasks können aktiviert oder deaktiviert werden. Deaktivierte Tasks werden nicht ausgeführt.

Zum Gestalten eines Tasks stehen die drei Basiskategorien

- 1) **Blöcke**
- 2) **Bedingungen** und
- 3) **Aktionen** zur Auswahl.

## 1) BLÖCKE

Die Blöcke bilden das Grundgerüst eines Tasks und dienen dazu, Bedingungen und Aktionen miteinander zu verbinden.

Dabei ist der **Sobald-Block** immer der Ausgangspunkt bzw. der **grundsätzliche Auslöser** (Trigger) des Tasks und bildet den Gesamtrahmen der Blöcke. Da systembedingt nicht permanent geprüft werden kann, ob ein Zustand besteht, ist der mit dem Sobald-Block festzulegende Auslöser an **den Eintritt eines Ereignisses** gekoppelt. Das bedeutet, dass der Task nicht solange aktiv bleibt, wie der als Auslöser festgelegte Zustand besteht, sondern der Task wird bei Eintreten dieses Zustands einmal durchlaufen.

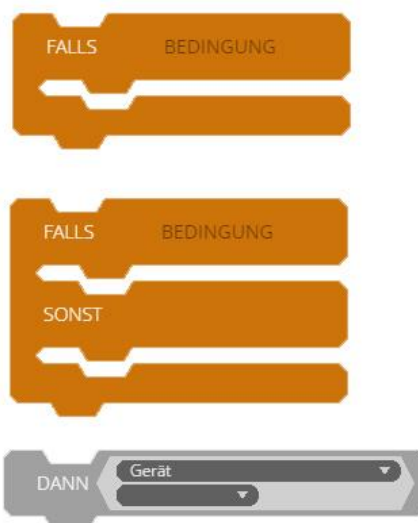
Da sie eine Grundvoraussetzung sind, können Sobald-Blöcke nicht gelöscht werden.



**i** Als **grundsätzliche Auslöser** zur Einbindung in den **Sobald-Block** können derzeit Geräte verwendet werden, die über folgende Gateways/Systeme in NEO integriert sind:

- ✓ AIO GATEWAYS
- ✓ Homematic CCU
- ✓ Logitech Harmony Hub
- ✓ Netatmo Wetterstation und Zusatzmodule, sowie Healthy Home Coach
- ✓ Doorbird

In den Sobald-Block können folgende weitere Blöcke eingebettet werden:





Der **Falls-Block** und der **Falls-Sonst-Block** können als erweiternde Bedingung für den Sobald-Block oder andere Falls- bzw. Falls-Sonst-Blöcke eingesetzt werden. Dabei sind Falls-Blöcke immer über eine logische **UND**-Verknüpfung mit dem übergeordneten Block verbunden.

Als **erweiternde Bedingungen** (in Falls/Falls-Sonst-Blöcken) oder als **Aktionen** können **alle** in NEO integrierten Geräte verwendet werden.

**i** *Wichtig:* Wird als Auslöser ein Gerätezustand festgelegt und als erweiternde Bedingung ebenfalls ein Gerätestatus gewählt (oder auch mehrere), so wird die erweiternde Bedingung überprüft, sobald der Auslösezustand eingetreten ist. Sie wird jedoch nicht etwa dauerhaft weiter geprüft, solange der Auslösezustand anhält.

### **Kurze Erläuterung der Blockfunktionalität**

#### → Sobald-Block bedeutet:

Sobald die zugeordneten Bedingungen eintreten, dann X

#### → Falls-Block bedeutet:

Sofern die zugeordneten Bedingungen eintreten, dann X

#### → Falls-Sonst-Block bedeutet:

Sofern die zugeordneten Bedingungen eintreten, dann X, wenn nicht, dann Y.

#### → Dann-Block bedeutet:

Über den **Dann-Block** können Aktionen an die Sobald-, Falls- und Falls-Sonst-Blöcke gekoppelt werden.

### Kombinieren von Blöcken:

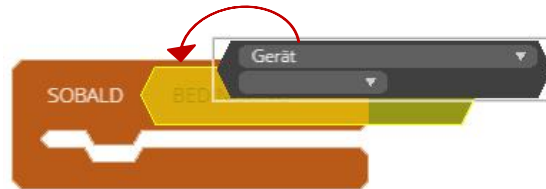
Blöcke lassen sich per Drag & Drop wie Puzzleteile miteinander verschachteln, indem sie an die gewünschte Position gezogen werden. Den äußeren Rahmen eines Tasks bildet immer der Sobald-Block. Alles, was sich außerhalb des Sobald-Blocks befindet, ist für den Task irrelevant (und wird beim Abspeichern gelöscht).

Werden Blöcke ineinander verschachtelt, so werden sie in der Abfolge von außen nach innen geprüft bzw. abgearbeitet.

### Zuweisen von Inhalten:

Zur Festlegung der Funktionalität von Blöcken müssen Inhalte zugewiesen werden. Je nach Art des Blocks können **Bedingungen** oder **Aktionen** hinzugefügt werden.

Das Hinzufügen erfolgt ebenfalls per Drag & Drop, indem eine Bedingung oder Aktion auf den jeweiligen Block gezogen wird.



## 2) BEDINGUNGEN

**Sobald-, Falls-** und **Falls-Sonst-**Blöcken können eine oder mehrere Bedingungen hinzugefügt werden.

Sobald-Block:

Sind dem Sobald-Block mehrere Bedingungen zugeordnet, sind diese immer mit **ODER** logisch verknüpft.

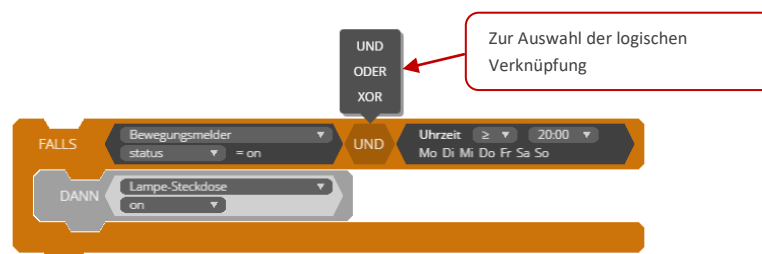
Falls- und Falls-Sonst-Block:

Sind diesen Blöcken mehrere Bedingungen zugeordnet, so können die Bedingungen über die drei unterschiedlichen logischen Verknüpfungen **UND**, **ODER** oder **XOR** miteinander kombiniert werden.

Dabei ist die XOR-Verknüpfung ein „exklusives ODER“. D.h.:

**ENTWEDER** wenn Bedingung A eintritt **ODER** wenn Bedingung B eintritt, dann X.

*(D.h. treten Bedingung A und B gleichzeitig ein, passiert nichts.)*



Folgende unterschiedlichen **Bedingungen** sind verfügbar:

## BEDINGUNGEN

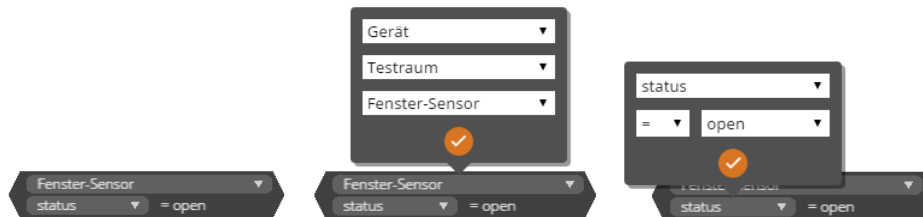
Klammern	
Gerätestatus	
Astro	
Zeit	
Intervall	
HTTP	
IR	

### 1. Klammern:

Die Klammer dient dazu mehrere Gerätestatusbedingungen miteinander direkt logisch zu verknüpfen, um das Ergebnis wiederum mit anderen Bedingungen verknüpfen zu können.

### 2. Gerätestatus:

Hier dient der Zustand eines Gerätes als Bedingung.



Durch Anklicken des oberen Feldes (gewünschtes Gerät) und des unteren Feldes (jeweiliger Zustand) kann ein Status als Bedingung definiert werden.

### 3. Astro:

Hier können Sie Astrodaten (Sonnenauf- & Sonnenuntergang) als Bedingung definieren.



Eine Astrobedingung kann nur einem Sobald-Block zugeordnet werden.

#### 4. Zeit:

Hier können Sie zeitabhängige Bedingungen definieren.

The screenshot shows a configuration window for a time-based condition. At the top, 'Uhrzeit' is set to 20:00. Below it, a row of days (Mo, Di, Mi, Do, Fr, Sa, So) is shown with checkmarks. The 'Zeitraum' section has 'Start' and 'Ende' both set to 21/02/2017. A confirmation button with a checkmark is at the bottom. Below the window, a summary bar shows 'Uhrzeit ≤ 20:00' and the days Mo, Di, Mi, Do, Fr, Sa, So.

#### 5. Intervall:

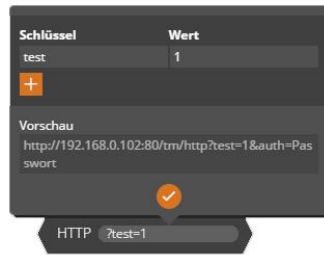
Hier können Sie ein Zeit-Intervall definieren, in dem der Task ausgelöst wird. So ist es möglich, in regelmäßigen Abständen den Task auszulösen, um bestimmte Statuszustände regelmäßig zu überprüfen, an die wiederum erweiterte Bedingungen geknüpft sind.

Wichtig: Die Intervall-Bedingung steht nur für den **Sobald**-Block zur Verfügung.

The screenshot shows a configuration window for an interval-based condition. At the top, 'Intervall' is set to 60 Sekunden. Below it, a row of days (Mo, Di, Mi, Do, Fr, Sa, So) is shown with checkmarks. The 'Auf Uhrzeit beschränken' section has 'Start' set to 00:00 and 'Ende' set to 23:59. The 'Auf Zeitraum beschränken' section has 'Start' and 'Ende' both set to 12/04/2017. A confirmation button with a checkmark is at the bottom. Below the window, a summary bar shows 'Intervall 60 00:00' and the days Mo, Di, Mi, Do, Fr, Sa, So.

#### 6. HTTP:

Hier können Sie einen HTTP-Request als Bedingung definieren, der dann als GET-Anfrage auf dem NEO SERVER aufgerufen werden kann.



### 7. IR:

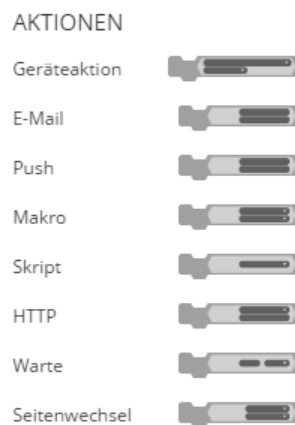
Mit der Bedingung IR wird es ermöglicht, IR-Signale als Auslöser zu verwenden. Dies funktioniert ausschließlich mit den AIO GATEWAYS V4/V4+/V5/V5+. Mit anderen IR-basierten Gateways wie dem Harmony Hub geht dies nicht, da diese die empfangenen IR-Signale nicht über das Netzwerk mitteilen.



### 3) Aktionen

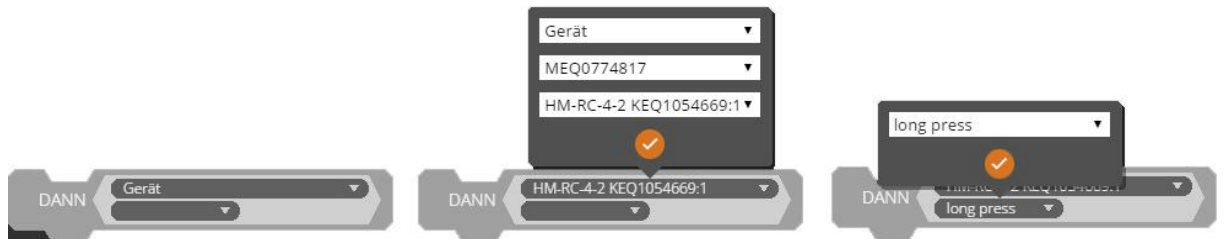
**Dann**-Blöcke führen immer **Aktionen** aus. Sollen beim Eintreten einer Bedingung mehrere Aktionen ausgelöst werden, so können mehrere Dann-Blöcke hintereinander in den Bedingungsblock eingefügt werden.

Folgende unterschiedliche Aktionen stehen für Dann-Blöcke zur Verfügung:



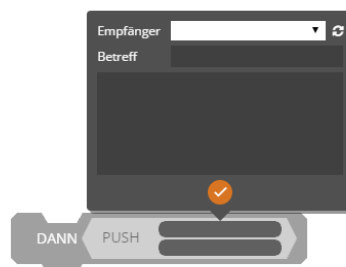
### 1. Geräteaktion:

Durch Anklicken des oberen Feldes (gewünschtes Gerät) und des unteren Feldes (jeweilige Aktion) kann die auszuführende Aktion definiert werden. Es sind hierbei alle im AIO CREATOR NEO angelegten Geräte auswählbar, die eine Aktion ausführen können.



### 2. Push:

Hier können Sie den Empfänger einer Push-Nachricht und einen Benachrichtigungstext als auszuführende Aktion eintragen. Die so erstellte Nachricht wird auf dem mediola Server hinterlegt und von dort verschickt.



### 2. E-Mail:

Hier können Sie eine E-Mail-Adresse und einen Benachrichtigungstext als auszuführende Aktion eintragen. Die so erstellte E-Mail wird auf dem mediola Server hinterlegt und von dort verschickt.



### 3. Makro:

Durch Anklicken des oberen Feldes (Makrogruppe) und des unteren Feldes (Makro) kann die auszuführende Aktion definiert werden. Es sind hierbei alle angelegten Makros auswählbar.



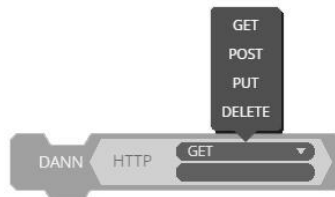
### 4. Skript:

Hier können Sie ein im Skripteditor gespeichertes Skript als auszuführende Aktion auswählen.



### 5. HTTP:

Hier können Sie einen HTTP-Request als Aktion definieren.



### 6. Warte:

Hier können Sie eine Pause mit einer definierten Dauer als auszuführende Aktion auswählen.



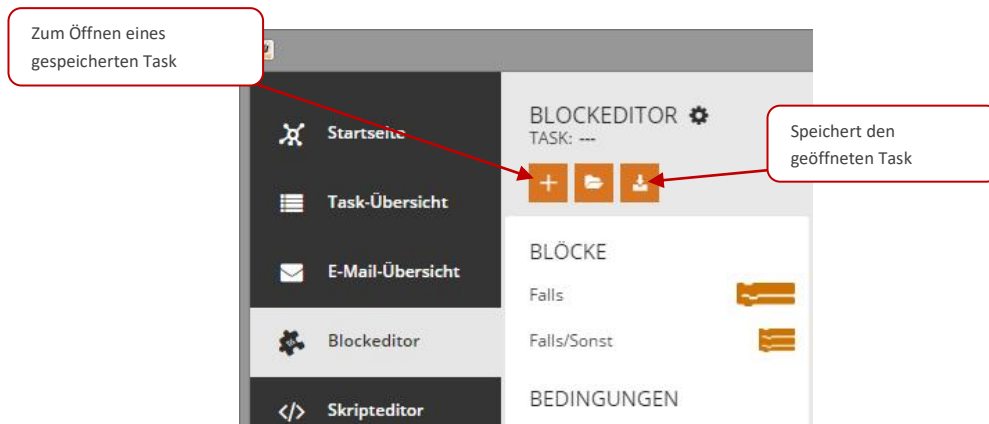
### 7. Seitenwechsel:

In diesem Block können Sie die Seite einer Fernbedienung definieren, zu der automatisch gewechselt wird, sobald der eingestellte Auslöse-Event eintritt.



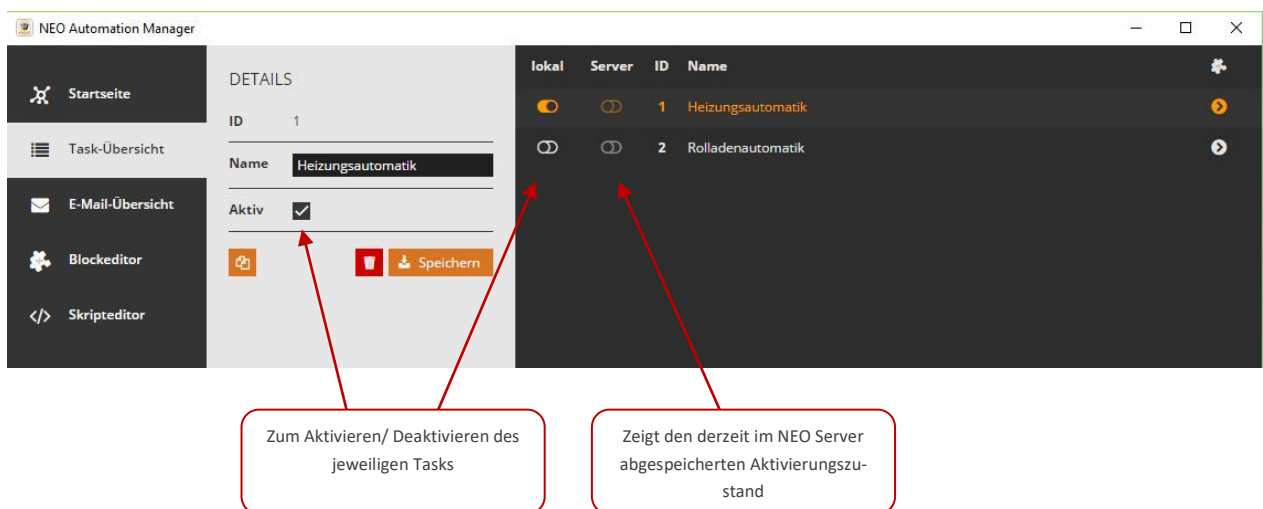
# AUFBAU UND BEISPIELE DES AUTOMATION MANAGERS

## Abspeichern und Öffnen von Tasks



## Task-Übersicht

In der Task-Übersicht werden alle gespeicherten Tasks angezeigt und können aktiv oder inaktiv geschaltet werden. Die beiden Spalten "lokal" und "Server" zeigen den jeweiligen Aktivierungsstatus eines Task an. Wenn der lokale Aktivierungs-Status geändert wurde, sind beide Status so lange asynchron, bis die Änderungen auf den NEO Server übertragen wurden.

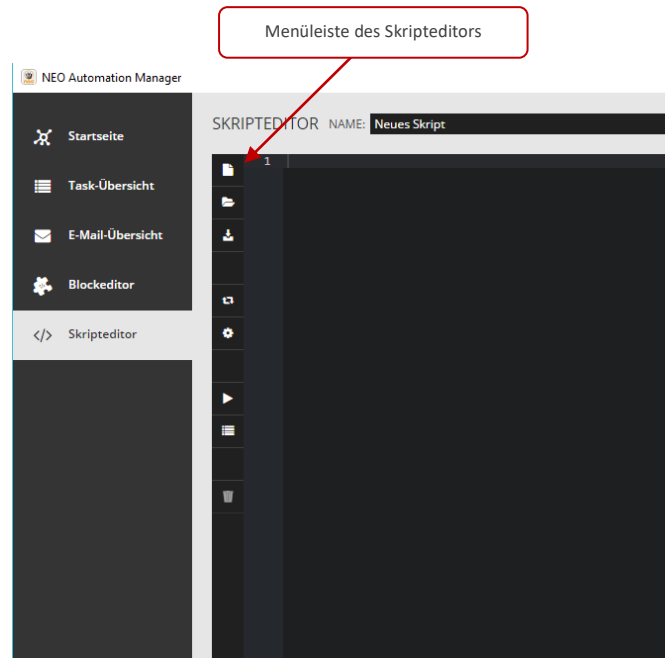


Außerdem kann hier ein Task zur Bearbeitung mit dem Blockeditor ausgewählt, der Name des Tasks geändert, der Task dupliziert oder gelöscht werden.

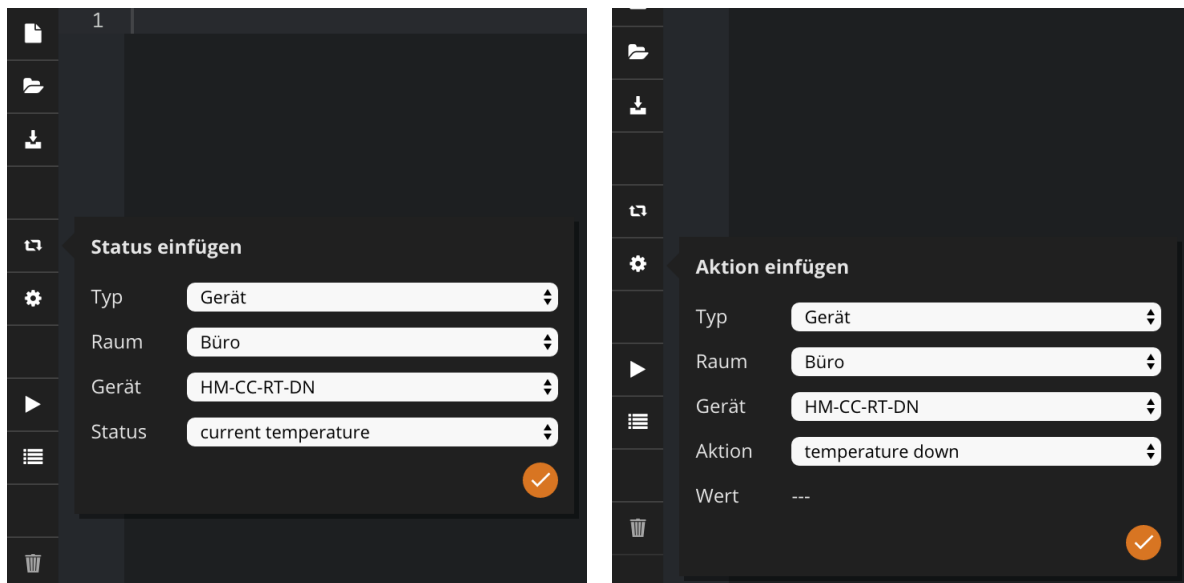


## Skripteditor

Im Skripteditor können individuelle Skripte mittels JavaScript geschrieben werden. In der Menüleiste stehen diverse Optionen (wie Speichern, Öffnen, Status/Aktion einfügen) zur Verfügung.



Die Code-Snippets für Statusabfragen und Aktionen lassen sich bequem über das Menü zusammenklicken.



**i** Wichtig: Die Aufrufe selbst erfolgen asynchron, weshalb mit dem Ergebnis erst in der Callback-Funktion gearbeitet werden kann.

Bereits gespeicherte Skripte können testweise auch im Skripteditor mit einem Klick auf das „Abspielen“-Symbol ausgeführt werden. In der ausklappbaren Liste der laufenden Skripte lassen sich diese auch vorzeitig wieder beenden, sollte ein Skript zum Beispiel zu lange benötigen.

## E-Mail-Übersicht

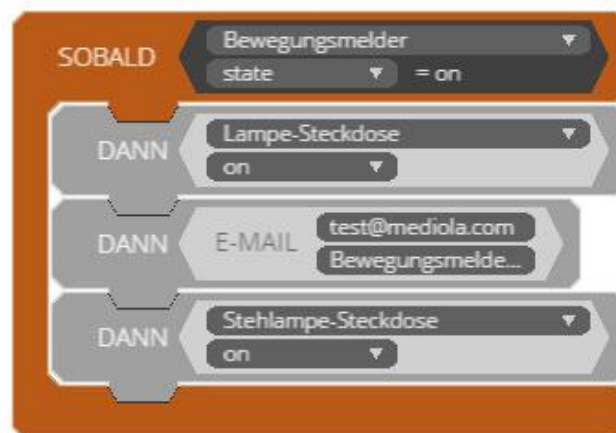
Hier werden Ihnen alle im Rahmen eines Tasks angelegten E-Mail-Benachrichtigungen angezeigt.

## Beispiele

Zur Verdeutlichung der Vorgehensweise sollen nachfolgend zwei Beispiele kurz erläutert werden. Dabei steht die Beschreibung der Kombinationsmöglichkeiten der Blöcke (und weniger die Sinnhaftigkeit des Beispiels) im Vordergrund.

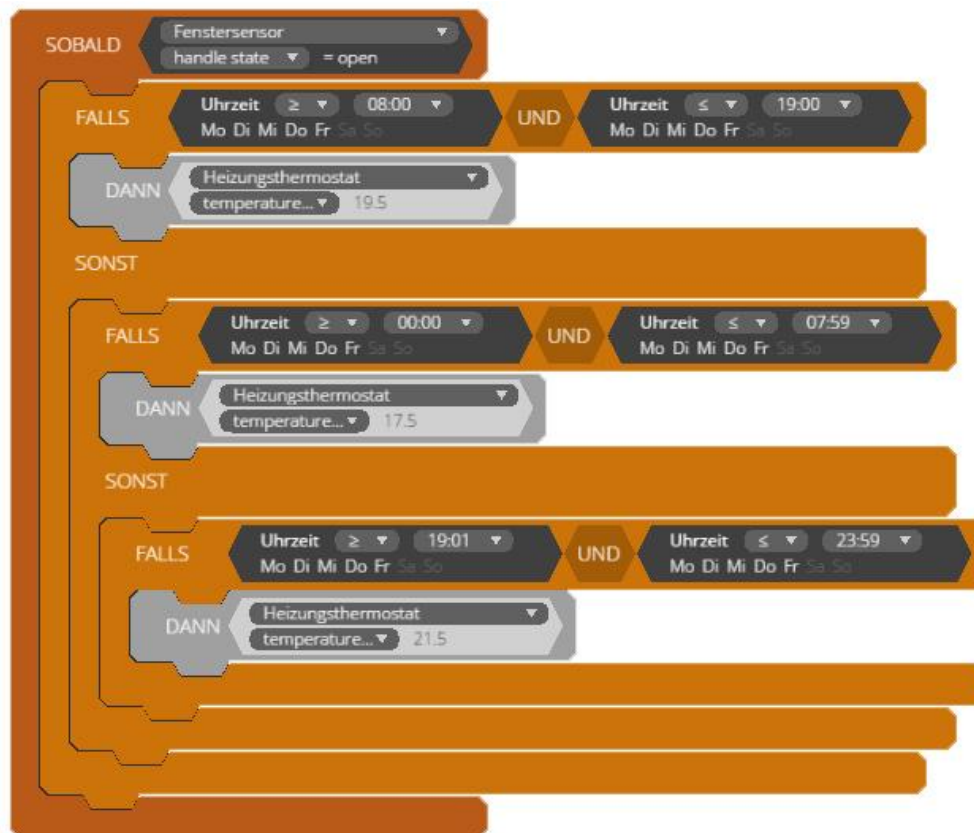
### Beispiel 1:

Wenn der Gartentor-Bewegungsmelder auslöst, wird zunächst eine Garten-Lampe angeschaltet sowie eine E-Mail-Benachrichtigung verschickt. Anschließend wird eine Stehlampe im Wohnzimmer eingeschaltet.



### Beispiel 2:

In diesem Beispiel werden zu dem zentralen Auslöser erweiternde Bedingungen eingefügt.



Sobald der Fenstersensor meldet, dass das Fenster geöffnet ist, wird das Heizungsthermostat von Montag bis Freitag in Abhängigkeit von der Tageszeit auf unterschiedliche Heiztemperaturen gestellt.

In der Zeit von 00:00 bis 07:59 Uhr auf 17,5 °C. In der Zeit von 08:00 bis 19:00 Uhr auf 19,5 °C. In der Zeit von 19:01 bis 23:59 Uhr auf 21,5 °C.

## UNTERSCHIEDE AUTOMATION MANAGER & TASK MANAGER PLUS

Der NEO Automation Manager ist ein kostenpflichtiges Plugin für den AIO CREATOR NEO, mit dem die in NEO eingebundenen Smart Home Geräte herstellerübergreifend automatisiert werden können.

Der Task Manager Plus ist Bestandteil des AIO GATEWAYS V5 Plus und ermöglicht die Erstellung von Tasks zur Automatisierung der an das AIO GATEWAY V5 Plus angelegten Geräte.

In der Handhabung und dem Aufbau mit grafischem Programmier-Frontend sind Task Manager Plus und NEO Automation Manager sehr ähnlich. Die Unterschiede sind:

Mit dem **Task Manager Plus** können nur Geräte automatisiert werden, **die an das AIO GATEWAY V5 Plus angelernt sind**, während die Automatisierung mit dem **NEO Automation Manager** alle **in NEO eingelesenen Geräte** beinhaltet, also auch solche Geräte, die sich nicht über das AIO GATEWAY V5 Plus ansteuern lassen.

Neben dem graphischen Programmier-Frontend mit per Drag & Drop setzbaren Blöcken bietet der **NEO Automation Manager** mit dem **Skripteditor** auch ein Profi-Frontend, indem mit Hilfe von JavaScript komplexere Skripte selbst erstellt werden. Der Task Manager Plus besitzt keinen Skripteditor.