



NEO Automation Manager

(Beta-Version)

KURZANLEITUNG

1. 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 System plattform- und technologieübergreifend automatisiert werden.

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

Zur Überwachung der Ausführung der mit dem NEO Automation Manager erzeugten Automatik-Programme dient der **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.

2. Installation des NEO SERVER

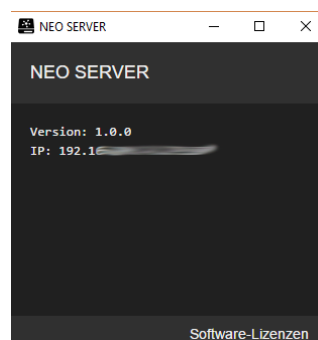
2.1. Windows und MacOS

Installieren Sie zunächst das Programm NEO SERVER für Ihr Betriebssystem. Die Installationsdateien können Sie unter den folgenden Links herunterladen.

macOS: Bei Interesse an einem Betatest bitte hier melden: beta@mediola.com

Windows: Bei Interesse an einem Betatest bitte hier melden: beta@mediola.com

Starten Sie nun das Programm. Es erscheint folgendes Programmfenster.



2.2. Raspberry PI

1. Installieren Sie zunächst das Programm "nodejs" LTS version (6.x.x) auf dem Raspberry PI

→ Gehen Sie in das "home" Verzeichnis des aktuellen Benutzer

```
cd
```

→ Laden Sie das Programm "nodejs" herunter und dekomprimieren Sie es anschließend

```
wget https://nodejs.org/dist/v6.9.2/node-v6.9.2-linux-armv6l.tar.gz
tar -xvf node-v6.9.2-linux-armv6l.tar.gz
cd node-v6.9.2-linux-armv6l
```

→ Kopieren Sie es in das Verzeichns `/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 `v6.9.2` erhalten.

2. Installieren Sie nun den "NEO SERVER" auf dem Raspberry PI

→ Gehen Sie in das "home" Verzeichnis des aktuellen Benutzer

```
cd
```

→ Laden Sie das Programm "NEO SERVER" herunter und dekomprimieren Sie es anschließend

```
wget DOWNLOADLINK
mkdir neo_server
cd neo_server
tar -xvf ../NEO+SERVER-2.0.0-linux.tar.gz
```

→ Starten Sie jetzt den NEO SERVER

```
sudo node automation.js
```

3. Starten Sie den "NEO SERVER" als System-Dienst

→ Erstellen Sie das Daemon Skript im Verzeichnis `/etc/init.d`

→ Zum Erstellen des Skriptes

```
cd /etc/init.d
sudo nano neo_server_daemon
```

→ Geben Sie nun das Folgende ein und passen dabei den `HOMEPATH` Parameter an das tatsächliche NEO SERVER Verzeichnis auf Ihrem Raspberry PI an. Anschließend bitte speichern und schließen.

```
#!/bin/sh
### BEGIN INIT INFO
# Provides: neo_server_daemon
# Required-Start: $network $remote_fs $syslog
# Required-Stop: $remote_fs $syslog
# Default-Start: 2 3 4 5
# Default-Stop: 0 1 6
# Short-Description: Start daemon at boot time for neo_server
# Description: Enable service provided by daemon.
### END INIT INFO
HOMEPATH=/home/pi/neo_server
PID=$(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
      $0 status
    fi
    ;;
  stop)
    if [ -z $PID]; then
      echo "neo_server is not running"
    else
      kill $PID
      echo "neo_server closed"
    fi
    ;;
  restart)
    if [ -z $PID]; then
      $0 start
    else
      $0 stop
      $0 start
    fi
    ;;
  status)
    if [ ! -z $PID]; then
      echo "neo_server is running PID $PID"
    else
      echo "neo_server is not running"
    fi
    ;;
  *)
    echo "Usage: $0 {start|stop|status|restart}"
    exit 1
    ;;
esac
exit 0
```

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

```
sudo chmod 0755 neo_server_daemon
sudo update-rc.d neo_server_daemon defaults
```

3. Integration des NEO SERVER in den AIO CREATOR NEO

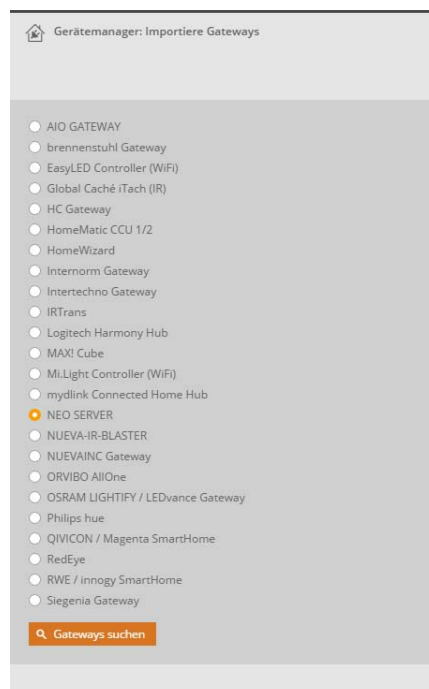
Haben Sie den NEO SERVER installiert und gestartet, müssen Sie ihn im Gerätemanager des AIO CREATOR NEO als neues Gateway einbinden. Dazu installieren Sie zunächst die Betaversion des AIO CREATOR NEO, die Sie hier für Ihr Betriebssystem downloaden können.

macOS: Bei Interesse an einem Betatest bitte hier melden: beta@mediola.com

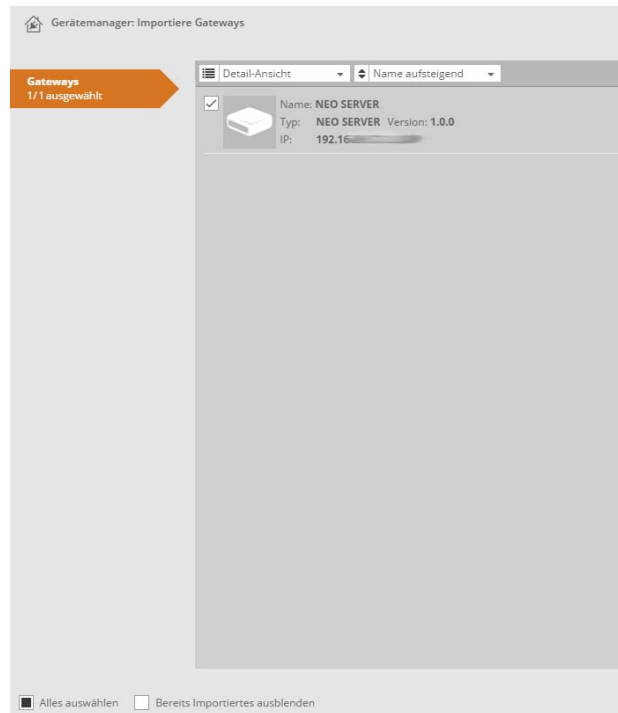
Windows: Bei Interesse an einem Betatest bitte hier melden: beta@mediola.com

Wichtig: Denken Sie bitte daran, Ihr laufendes System vorher zu sichern

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



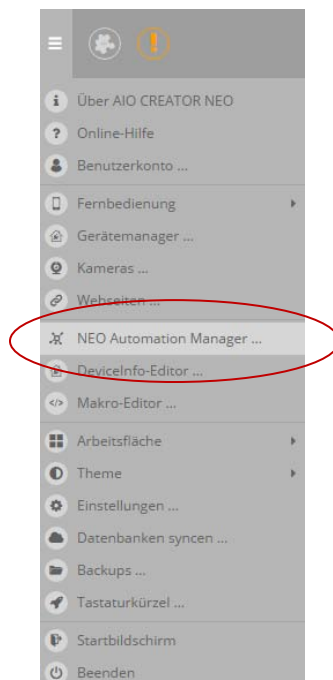
Im folgenden Auswahlménü selektieren Sie bitte "NEO SERVER" und klicken anschließend auf "Gateways suchen".



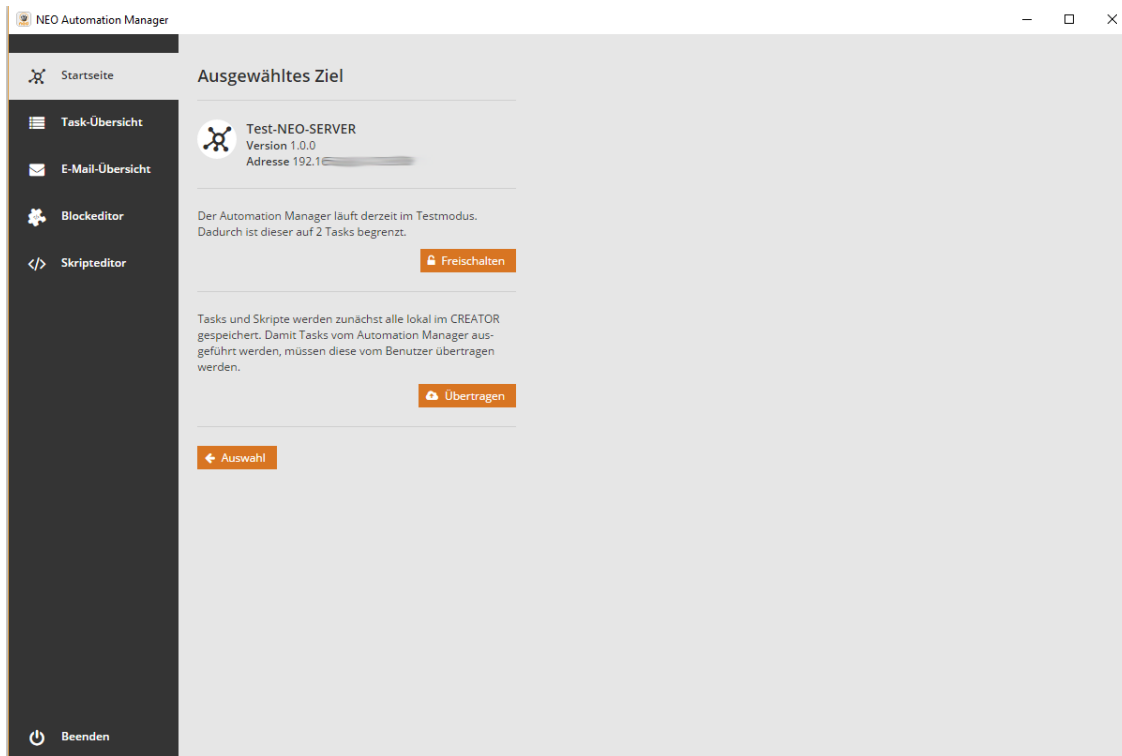
Nun wird der NEO SERVER angezeigt und kann durch Anklicken der "Importieren"-Taste als neues Gateway in den Geräte-Manager importiert werden. Anschließend wird der NEO SERVER als Gateway im Gateways-Tab aufgelistet und man kann dort einen individuellen Namen vergeben. Haben Sie diesen Vorgang abgeschlossen,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

4. Erzeugen von Tasks mit dem NEO Automation Manager

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



Auf der Startseite sehen Sie den zuvor integrierten NEO SERVER als ausgewähltes Ziel zur Übertragung der im Automation Manager erzeugten Tasks.



Hinweis:

Im Testmodus können bis zu zwei Tasks erstellt werden. 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 übertragen werden. Nutzen Sie zum Übertragen die "Übertragen"-Taste auf der Startseite. Nach der Übertragung wird die Ausführung der Tasks allein vom NEO SERVER überwacht und durchgeführt (der AIO CREATOR NEO wird dazu nicht benötigt).

4.1. Erstellen von Tasks

Tasks werden grundsätzlich im Blockeditor erzeugt. Im Skripteditor erstellte Skripte können ebenfalls über den Blockeditor als Aktion in einen Task integriert werden.

4.2. 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 Task stehen die drei Basiskategorien "**Blöcke**", "**Bedingungen**" und "**Aktionen**" zur Auswahl.

4.2.1. Blöcke

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

Dabei ist der **Wenn-Block** immer der Ausgangspunkt bzw. der **grundsätzliche Auslöser** (Trigger) des Task und bildet den Gesamtrahmen (der Blöcke). Da systembedingt nicht permanent geprüft werden kann, ob ein Zustand besteht, ist der mit dem Wenn-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 quasi einmalig "gestartet".

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

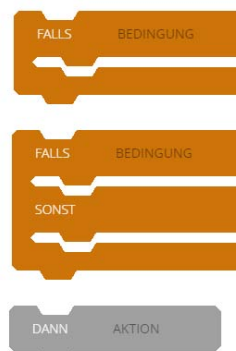


Hinweis:

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

- AIO GATEWAYS
- Homematic CCU
- Logitech Harmony Hub
- Netatmo Wetterstation

In den Wenn-Block können folgende weitere Blöcke eingebettet werden.



Der **Falls-Block** und der **Falls-Sonst-Block** können als erweiternde Bedingung für den Wenn-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/ Systeme verwendet werden.

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:

→ Wenn-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 Wenn-/ Falls- und Falls-Sonst-Blöcke gekoppelt werden.

Kombinieren von Blöcken:

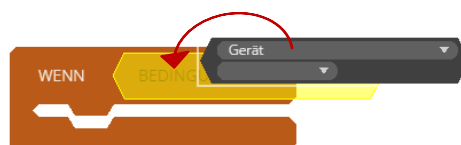
Blöcke lassen sich per Drag'n Drop wie Puzzleteile miteinander verschachteln, indem sie an die gewünschte Position gezogen werden. Den äußeren Rahmen eines Task bildet immer der Wenn-Block. Alles, was sich außerhalb des Wenn-Blocks befindet, ist für den Task irrelevant (und wird beim Abspeichern gelöscht).

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'n Drop, indem eine Bedingung oder Aktion auf den jeweiligen Block gezogen wird.



4.2.2. Bedingungen

Wenn-/ Falls- und **Falls-Sonst-**Blöcken können eine oder mehrere Bedingungen hinzugefügt werden.

Wenn-Block:

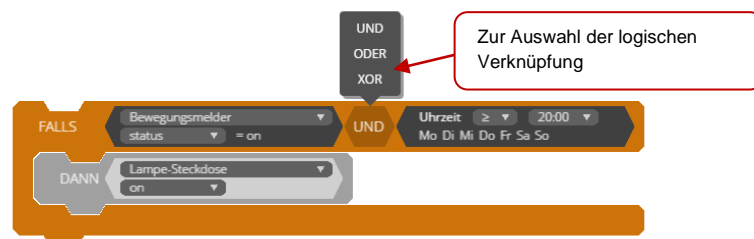
Sind dem Wenn-Block mehrere Bedingungen zugeordnet, sind diese immer mit **ODER** logisch verknüpft.

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



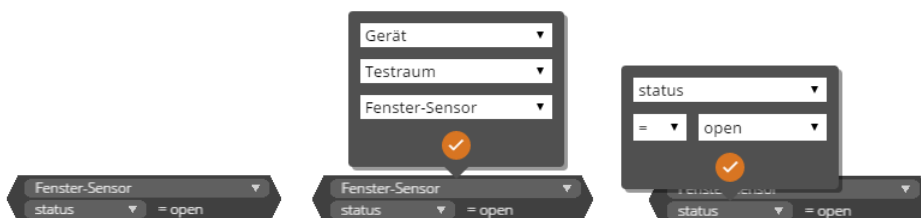
Folgende unterschiedlichen **Bedingungen** sind verfügbar:

BEDINGUNGEN

- Klammern
- Gerätestatus
- Astro
- Zeit
- HTTP

1. 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. Es sind hierbei alle im Gerätemanager angelegten Geräte auswählbar, die einen Status liefern.

2. Astro:

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

Eine Astrobedingung kann nur einem Wenn-Block zugeordnet werden.

3. Zeit:

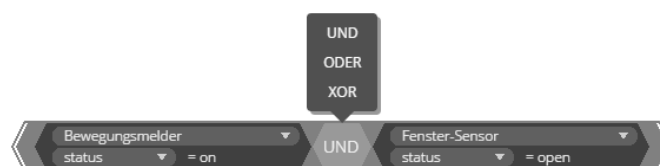
Hier können Sie zeitabhängige Bedingungen definieren.

4. HTTP:

Hier können Sie einen HTTP-Request als Bedingung definieren.

5. Klammer:

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



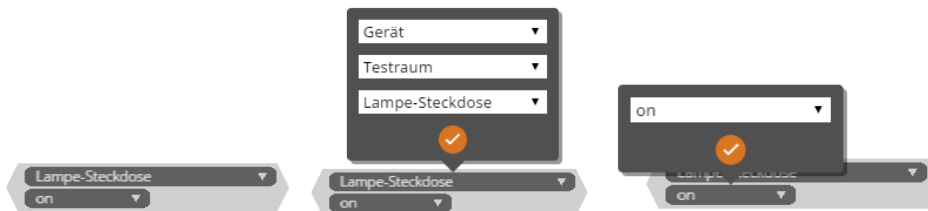
4.2.3. Aktionen

Dann-Blöcke führen immer **Aktionen** aus. Die Aktionen werden dem Block per Drag'n Drop zugewiesen. 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 Gerätemanager angelegten Geräte auswählbar, die eine Aktion ausführen können.

2. E-Mail:

Hier können Sie eine E-Mail-Adresse und einen Benachrichtigungstext als auszuführende Aktion eintragen.



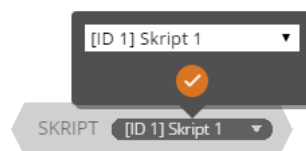
3. Makro:



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

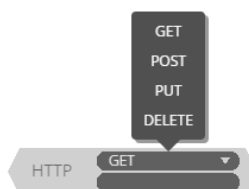
4. Skript:

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



5. HTTP:

Hier können Sie ein HTTP-Befehl als auszuführende Aktion auswählen.

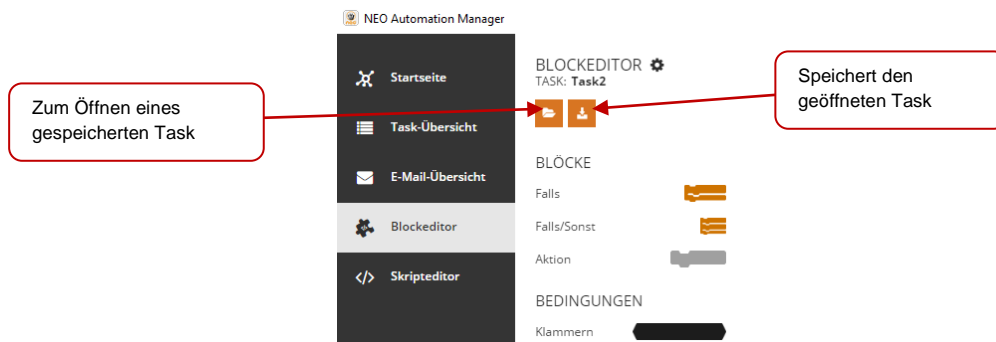


6. Warte:

Hier können Sie eine Pausendauer als auszuführende Aktion auswählen.

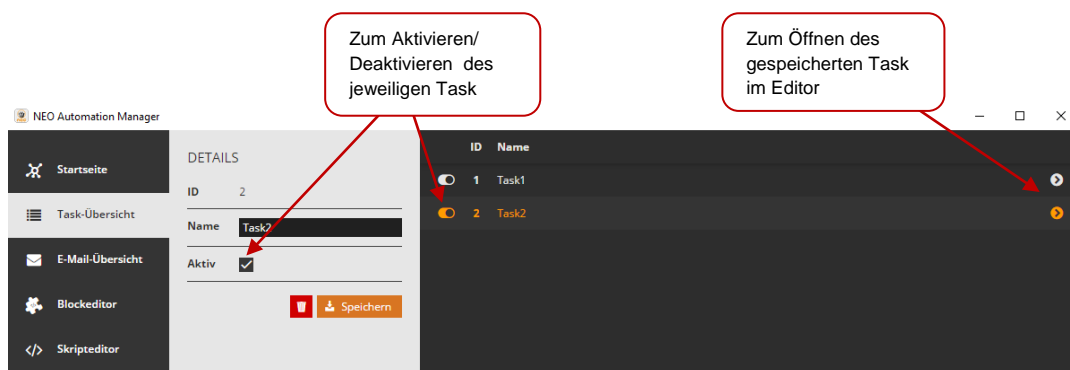


4.3. Abspeichern und Öffnen von Tasks



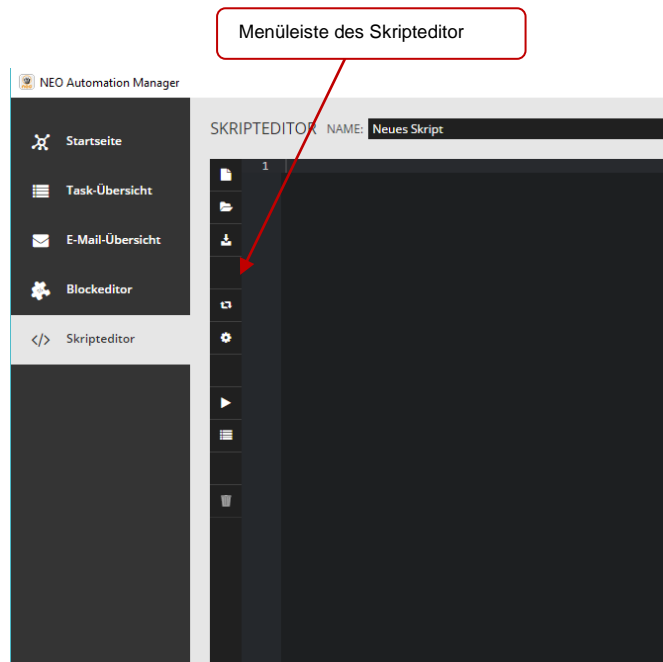
4.4. Task Übersicht

In der Task Übersicht werden alle gespeicherten Tasks angezeigt und können aktiv oder inaktiv geschaltet werden. Außerdem kann hier ein Task zur Bearbeitung mit dem Blockeditor ausgewählt werden.



4.5. Skripteditor

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



Gespeicherte Skripte können als Aktion im Blockeditor in einen Task integriert werden.

4.6. E-Mail Übersicht

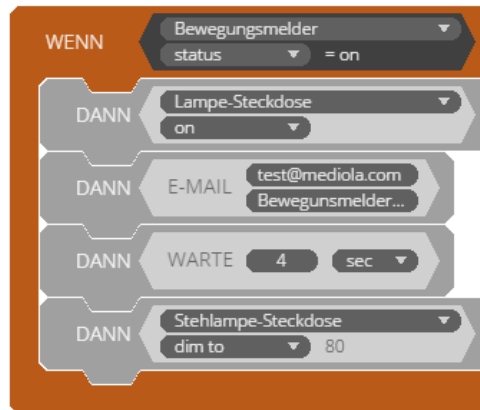
Hier werden Ihnen alle im Rahmen eines Task ausgelösten E-Mail-Benachrichtigungen angezeigt.

4.7. 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.

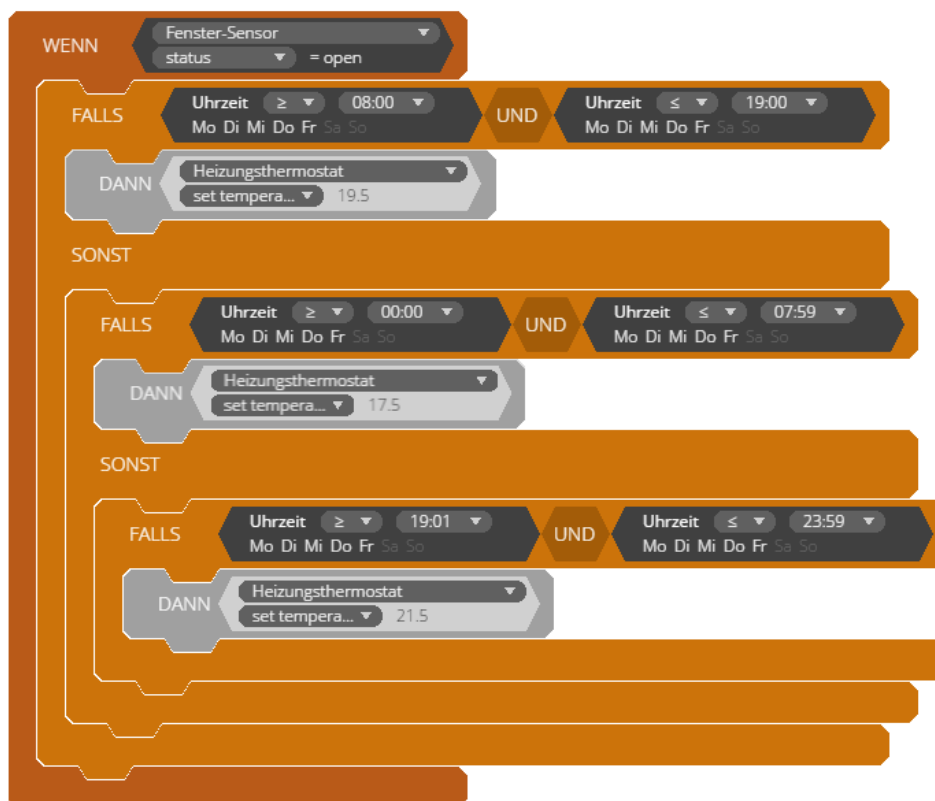
Beispiel 1:

Wenn der Gartentor-Bewegungsmelder auslöst, wird zunächst eine Garten-Lampe angeschaltet sowie eine E-Mail-Benachrichtigung verschickt. Nach einer Pause von 4 Sekunden wird schließlich eine Lampe im Haus auf 80% gedimmt.



Beispiel 2:

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



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