

Integracja z NETIO PowerPDU

Numer dokumentu: PO-182 Wersja: 1.0 Data publikacji: 6 października 2022

Wstęp

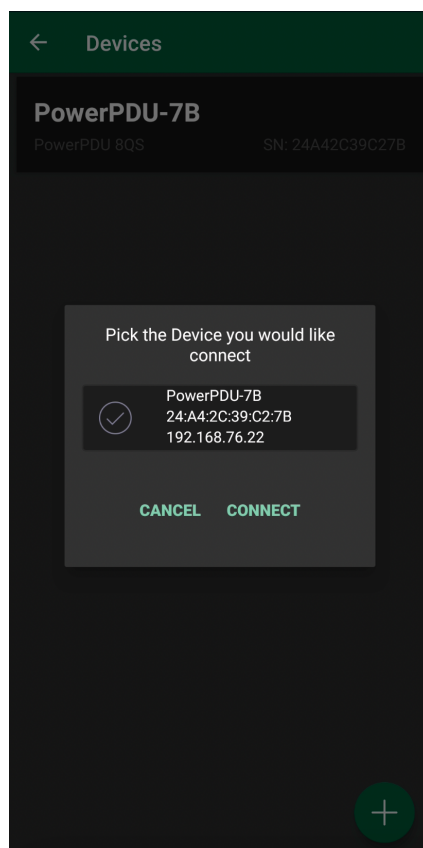
Ampio udostępnia możliwość integracji poprzez platformę Node-RED oraz daje do dyspozycji bibliotekę bloków ułatwiających różnego rodzaju integracje. Podstawy integrowania Ampio przy pomocy Node-RED są opisane w poradniku [Integracja systemu Ampio z Node-RED](#).

Ten poradnik wyjaśnia kroki integrowania instalacji Ampio z inteligentnymi listwami zarządzającymi zasilaniem (ang. PDU) firmy NETIO. Na potrzeby tego dokumentu użyto modelu NETIO PowerPDU 8QS.

Podłączanie listwy NETIO

Pierwszym krokiem jest podłączenie urządzenia NETIO do źródła prądu oraz do kabla Ethernet w lokalnej sieci.

Następnie, należy znaleźć adres IP podłączonego urządzenia. W tym celu należy ściągnąć aplikację mobilną NETIO i podłączyć telefon komórkowy z zainstalowaną aplikacją do tej samej sieci, do której wpięta jest listwa. W menu aplikacji przejdź do opcji *Devices* i kliknij *Find device*. Aplikacja zwróci informacje o podłączonym urządzeniu NETIO wraz z jego adresem IP.



Zapisz adres IP, żeby później móc wykorzystać go w programie Node-RED.

Netio.Json

NETIO PowerPDU udostępnia szereg narzędzi pomiarowych dla każdego ze swoich wyjść. W celu zintegrowania tej funkcjonalności ze instalacją Ampio, trzeba uzyskać dostęp do konfiguracji JSON urządzenia NETIO. Można to osiągnąć poprzez wklejenie następującego URL w oknie przeglądarki: **<http://device-IP/netio.json>**. Powinniśmy zobaczyć informacje przedstawione poniżej.

```
{
  "Agent": {
    "Model": "805",
    "DeviceName": "PowerPDU-7B",
    "MAC": "24:A4:2C:39:C2:7B",
    "SerialNumber": "24A42C39C27B",
    "JSONVer": "2.4",
    "Time": "2022-07-19T08:57:40+01:00",
    "Uptime": 65178,
    "Version": "3.2.2",
    "OemID": 600,
    "VendorID": 0,
    "NumOutputs": 8,
    "NumInputs": 1,
    "GlobalMeasure": {
      "Voltage": 238.33,
      "TotalCurrent": 0,
      "OverallPowerFactor": 1.00,
      "TotalPowerFactor": 1.00,
      "OverallPhase": 0,
      "TotalPhase": 0,
      "Frequency": 50.05,
      "TotalEnergy": 0,
      "TotalReverseEnergy": 0,
      "TotalEnergyNR": 0,
      "TotalReverseEnergyNR": 0,
      "TotalLoad": 0,
      "EnergyStart": "1970-01-01T00:00+01:00",
      "R": 0
    },
    "Outputs": [
      {
        "ID": 1,
        "Name": "Power output 1",
        "State": 0,
        "Action": 6,
        "Delay": 5000,
        "Current": 0,
        "PowerFactor": 1.00,
        "Phase": 0.00,
        "Energy": 0,
        "ReverseEnergy": 0,
        "EnergyNR": 0,
        "ReverseEnergyNR": 0,
        "Load": 0
      },
      {
        "ID": 2,
        "Name": "Power output 2",
        "State": 0,
        "Action": 6,
        "Delay": 5000
      },
      {
        "ID": 3,
        "Name": "Power output 3",
        "State": 0,
        "Action": 6,
        "Delay": 5000
      },
      {
        "ID": 4,
        "Name": "Power output 4",
        "State": 0,
        "Action": 6,
        "Delay": 5000
      },
      {
        "ID": 5,
        "Name": "Power output 5",
        "State": 0,
        "Action": 6,
        "Delay": 5000
      },
      {
        "ID": 6,
        "Name": "Power output 6",
        "State": 0,
        "Action": 6,
        "Delay": 5000
      },
      {
        "ID": 7,
        "Name": "Power output 7",
        "State": 0,
        "Action": 6,
        "Delay": 5000
      },
      {
        "ID": 8,
        "Name": "Power output 8",
        "State": 0,
        "Action": 6,
        "Delay": 5000
      }
    ],
    "Inputs": [
      {
        "ID": 1,
        "Name": "Input 1",
        "State": 0,
        "S0Counter": 0
      }
    ],
    "PAB": [
      {
        "Type": "RANGE",
        "Name": "PAB_RANGE_1",
        "Enabled": false,
        "In": false
      },
      {
        "Type": "ZONES",
        "Name": "PAB_ZONES_1",
        "Enabled": false,
        "Zone": 0
      }
    ],
    "Watchdogs": [
      {
        "Name": "default_pinger",
        "Enabled": false,
        "Fail": false,
        "LastStatus": false,
        "Timestamp": 0
      }
    ],
    "Rules": [
      {
        "Name": "Rule1",
        "Enabled": false,
        "Result": false
      },
      {
        "Name": "Rule2",
        "Enabled": false,
        "Result": false
      },
      {
        "Name": "Rule3",
        "Enabled": false,
        "Result": false
      }
    ]
  }
}
```

Więcej użytecznych informacji dotyczących struktury JSON w kontekście produktów NETIO można znaleźć na [stronie internetowej](#) producenta inteligentnych listw zarządzających zasilaniem.

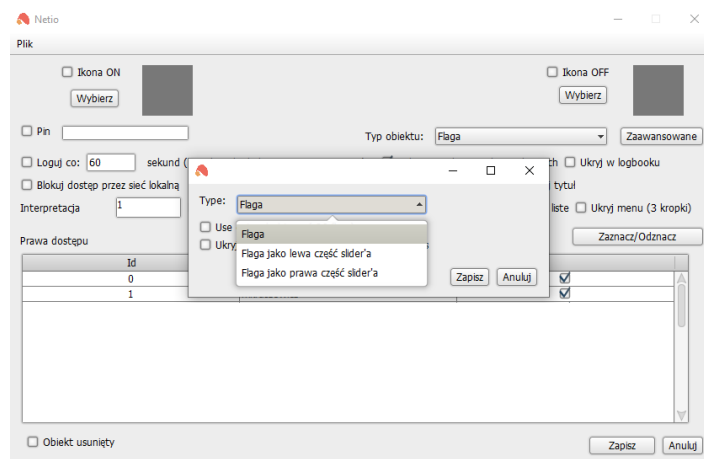
Dodawanie obiektów w Smart Home Manager

Aby zintegrować NETIO PowerPDU z Ampio, trzeba przeprowadzić dwie osobne konfiguracje w programie Smart Home Manager, tj. jedną do pozyskiwania statusu urządzenia NETIO, a drugą dla urządzenia kontrolującego.

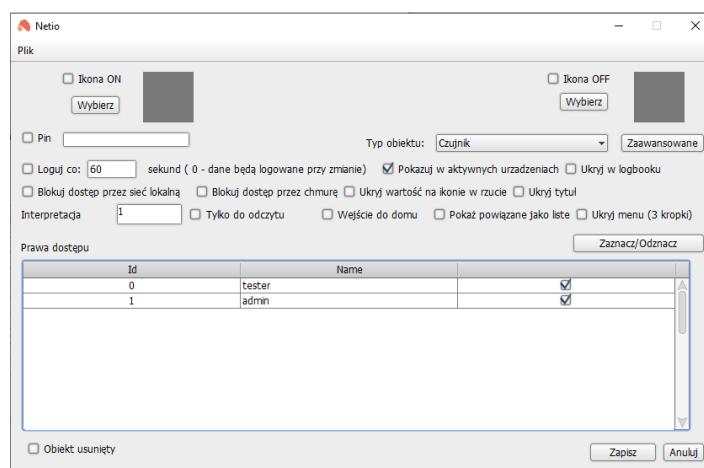
Najpierw otwórz Smart Home Manager przejdź do zakładki "Obiekty" w menu po lewej stronie ekranu. Następnie, na górnym pasku okienka kliknij "Dodaj", aby dodać nowe obiekty. Liczba obiektów, które należy stworzyć powinna odpowiadać ilości wyjść listwy NETIO, które będą wykorzystywane oraz liczbie wartości, np. napięcie, czy prąd, które będziemy chcieli mierzyć. W tym momencie użyteczna może być zmiana Opisu stworzonych obiektów na np. *Netio wyjście 1*, *Netio wyjście 2*, *Napięcie*, itd.

1403/200:	Napięcie	0	1 Wartość 32 bit	Ustawienia
1407/200:	Prąd	0	2 Wartość 32 bit	Ustawienia
1408:cefs:PANEL MARKETING	ON/OFF KLIMA	0	1 Flaga	Ustawienia
1409:cefs:PANEL MARKETING	Grzanie KLIMA	0	2 Flaga	Ustawienia
1410:cefs:PANEL MARKETING	Chłodzenie KLIMA	0	3 Flaga	Ustawienia
1411:cefs:PANEL MARKETING	Wyjście 4	0	4 Flaga	Ustawienia
1412:cefs:PANEL MARKETING	Wyjście 5	0	5 Flaga	Ustawienia
1413:cefs:PANEL MARKETING	Wyjście 6	0	6 Flaga	Ustawienia
1414:cefs:PANEL MARKETING	Wyjście 7	0	7 Flaga	Ustawienia
1415:cefs:PANEL MARKETING	Wyjście 8	0	8 Flaga	Ustawienia
1416/200:	Energia	0	9 Wartość 32 bit	Ustawienia
1417/200:	Częstotliwość	0	10 Wartość 32 bit	Ustawienia

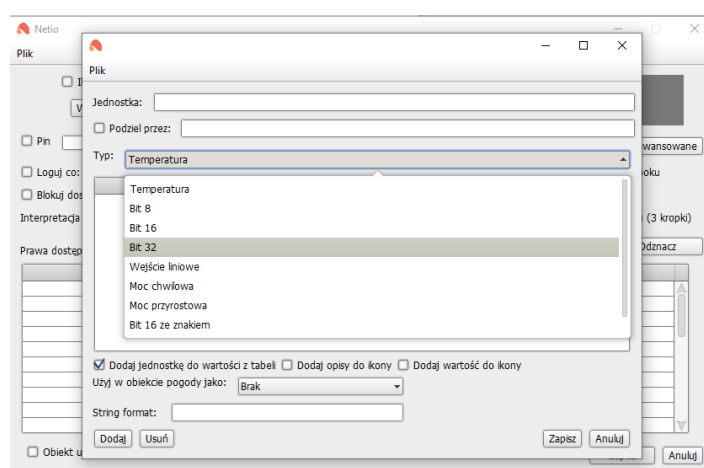
Następnie, otwórz ustawienia każdego ze stworzonych obiektów wyjść i ustaw ich typ obiektu na "Flaga".



W podobny sposób, dla obiektów stworzonych w celu wykonywania pomiarów wybierz typ “Czujnik”, tak jak pokazano to poniżej.



W drugiej kolejności, kliknij przycisk “Zaawansowane” i w nowo otwartym okienku zlokalizuj “Typ” (domyślnie ustawiony na temperaturę) i zmień go na “Bit 32”.



Nie zamykaj Smart Home Managera - już za chwilę będziemy się odnosić w Node-RED do numerów ustawionych flag.

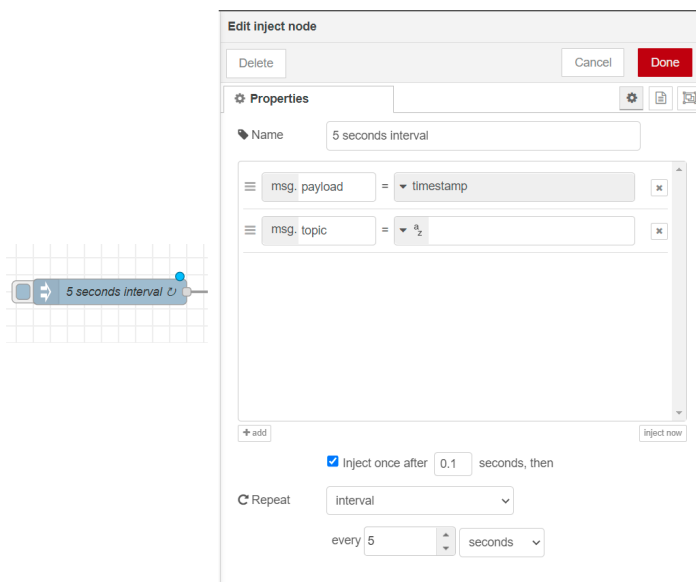
Konfiguracja Node-RED

Otwórz Node-RED i zaloguj się za pomocą swoich danych logowania.

Konfiguracja *inject node*

Bloczki typu *inject node* pozwalają na wstrzykiwanie wiadomości do flow albo poprzez kliknięcie przycisku na bločku, albo przez ustawienie interwałów czasowych pomiędzy kolejnymi wiadomościami. Do celów integracji z NETIO będziemy korzystali z drugiej opcji.

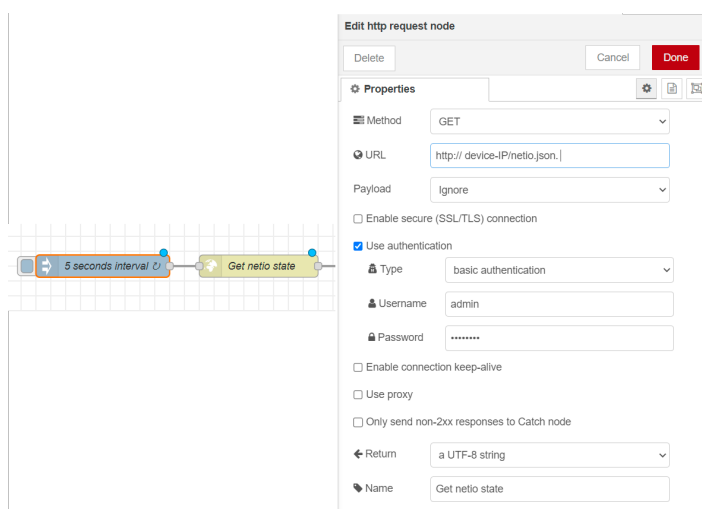
Zaczynamy od przeciągnięcia *inject node* z palety na pulpit Node-RED. Następnie, wchodzimy w edycję bločka i ustawiamy 5-sekundowe interwały, które pozwolą nam na odczytywanie stanów wyjść oraz pozyskiwanie informacji na temat zużycia prądu co 5 sekund.



Należy pamiętać, aby odpowiednio nazywać konfigurowane bloczki, żeby później móc się łatwiej między nimi poruszać.

Konfiguracja *HTTP request node*

Kolejnym bločkem, który będziemy dodawać jest *HTTP request node*. Otwórz jego właściwości i ustaw *Method* na *GET* oraz wklej **http://device-IP/netio.json** w pole *URL*. Następnie, ustaw *Payload* na *Ignore*. Użyj także podstawowej formy uwierzytelnienia (basic authentication). Tutaj należy podać nazwę użytkownika i hasło konta NETIO.



JSON node

W kolejnym kroku będziemy dodawać bloczek *JSON node*, który pozwala na konwersję pomiędzy JSON, a JavaScript. *JSON node* domyślnie rozpoznaje, co zostało mu zadane, i w którą stronę musi wykonać konwersję.



Konfiguracja *function node*

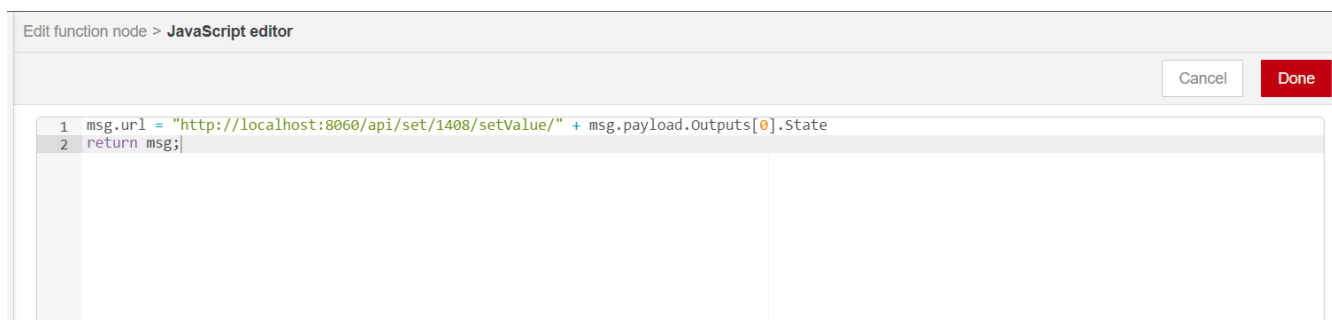
Bloczek *function node* umożliwia wykonanie kodu JavaScript na podstawie wiadomości, które są przez ten bloczek wysyłane. Posłuży on nam do uzyskania informacji o statusach wyjść urządzenia NETIO. Zaczniemy od dodania bloczków *function node* na pulpit i odpowiedniego ich skonfigurowania, co zaprezentowano poniżej.

W zakładce *On Message* tab, dodaj:

```

msg.url = (
  "http://localhost:8060/api/set/id/setValue/"
  + msg.payload.Outputs[0].State
);
return msg;

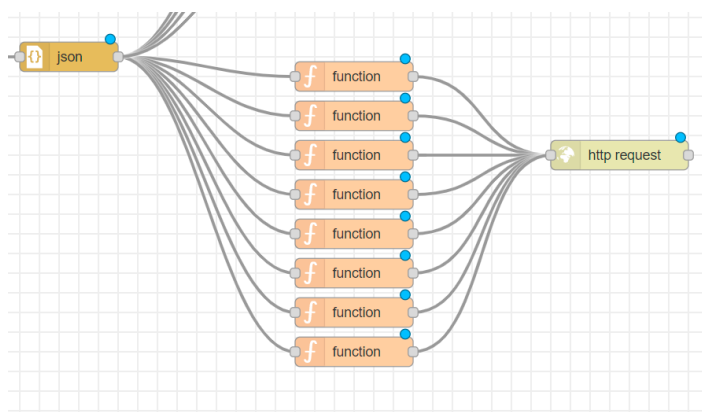
```



Powyższy przykład demonstruje pozyskanie informacji o Wyjściu nr 1 listy NETIO. W ten sam sposób, wpisanie `Outputs[1]` w kodzie zwróci nam informację o Wyjściu nr 2, itd.

Konfiguracja *HTTP request node* - jednostki pomiarowe

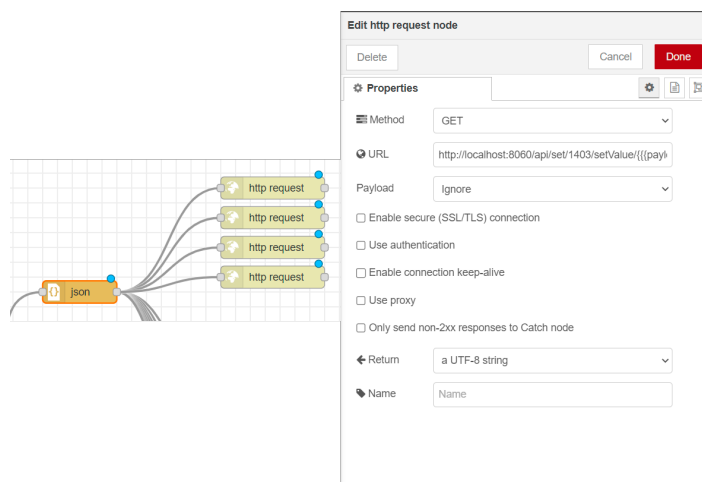
Ten bloczek powinien być dodany w tym momencie, jeżeli chcemy mieć możliwość odczytu takich wartości, jak napięcie, pobór prądu etc. przy pomocy urządzenia NETIO zintegrowanego z Ampio.



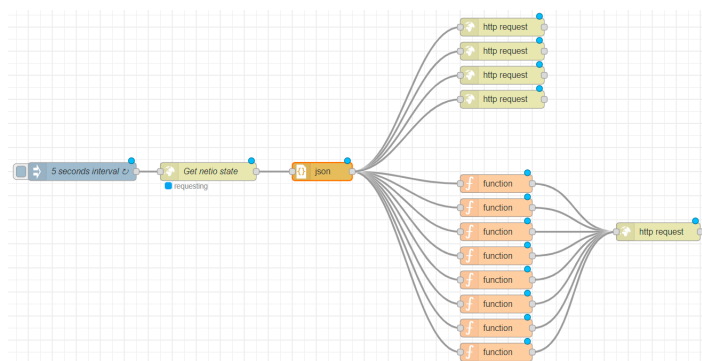
Konfiguracja przebiega podobnie do kroków opisanych w poprzedniej sekcji, za wyjątkiem:

1. Tym razem nie będziemy korzystać z uwierzytelnienia
2. W polu URL należy wkleić następujący adres URL:

```
__http://localhost:8060/api/set/1403/setValue/{{payload.GlobalMeasure.Voltage}}__
```

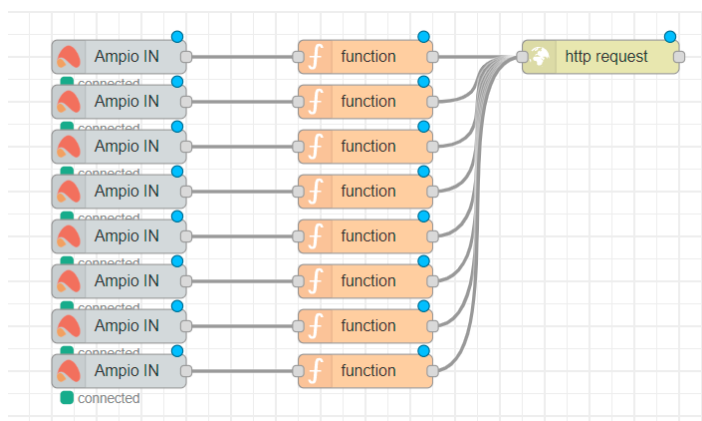


Powyższy przykład URL zwróci nam pomiar napięcia, ale można użyć w URL innych kluczowych zwrotów z konfiguracji NETIO JSON, aby uzyskać odczyty innych miar elektryczności, takich jak zużycie prądu, czy obciążenie, etc.



Blocek *Ampio IN*

Ostatnim etapem konfiguracji Node-RED, która pozwoli nam na zmianę stanów wyjść NETIO i ich kontrolowanie z poziomu aplikacji Ampio UNI, jest dodanie blozków *Ampio IN* oraz *function node*.



W bločku *Ampio IN* należy wprowadzić numer flagi, która wcześniej została przypisana w programie Smart Home Manager.

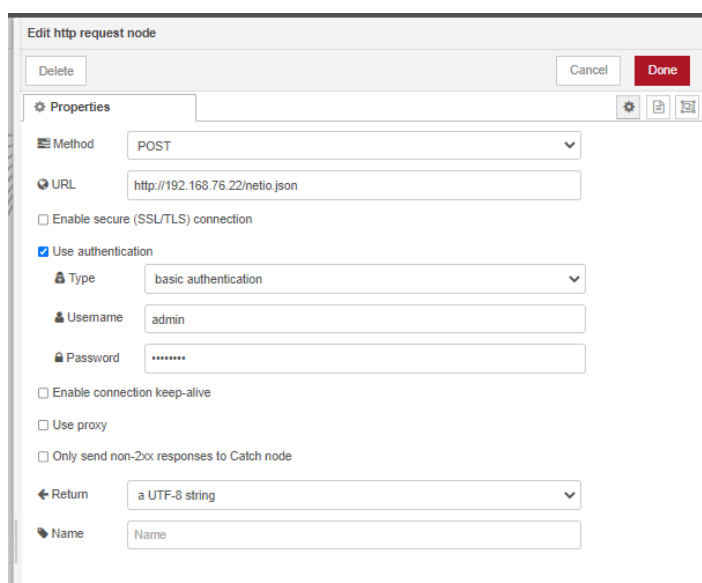
Natomiast w bločku *function node* należy wpisać następujący kod w zakładce *OnMessage*:

```
//msg.payload = "{ \"Outputs\": [{ \"ID\": 1, \"Action\": 4 }]}"
```

```
msg.payload = { Outputs: [{ ID: 1, Action: msg.payload }] };  
return msg;
```

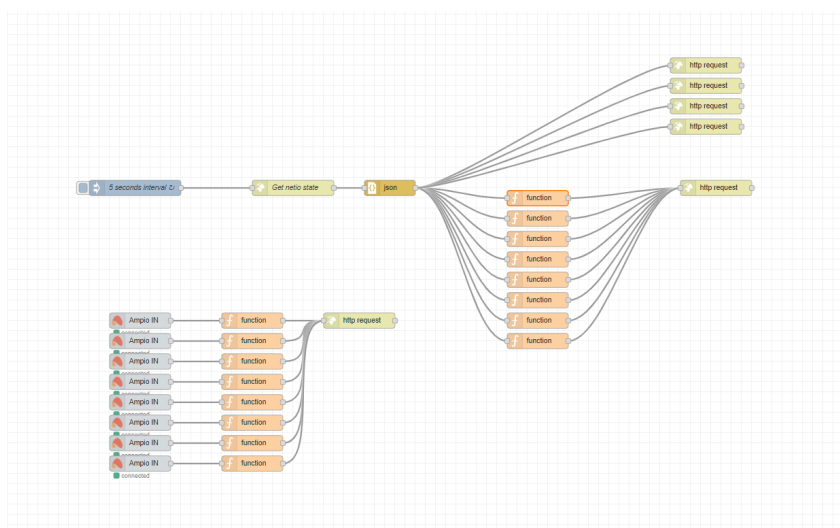
Wartość "ID" będzie zmieniana w każdym bločku *function node* dla każdego wyjścia, przykładowo, Wyjście nr 1 otrzyma "ID: 1", Wyjście 2 - "ID: 2, etc.

Pozostaje nam skonfigurowanie bločku *http request node* w taki sam sposób, jak w poprzednich krokach z dodatkowym zaznaczeniem podstawowego uwierzytelnienia (basic authentication) i podaniem tam danych logowania NETIO.



Skończona konfiguracja Node-RED

Proces konfiguracji jest już zakończony. Poniższy obrazek prezentuje pulpit Node-RED z kompletną konfiguracją listwy NETIO PowerPDU integrowanej z systemem Ampio.



Ostatnim krokiem jest kliknięcie *Deploy* w aplikacji Node-RED, co zapisze konfigurację na module z rodziny M-SERV w danej instalacji.

Ampio UNI

Aby sfinalizować proces integracji i móc obserwować jego owoce w aplikacji Ampio UNI, należy wrócić do programu Smart Home Manager i stworzyć nową grupę "Netio" w zakładce *Groupowanie* po lewej stronie ekranu.

Następnie, należy wybrać wszystkie obiekty stworzone w kroku Dodawanie obiektów w Smart Home Manager do nowo stworzonej grupy Netio.

Po zapisaniu, można rozpocząć zarządzanie listwą NETIO z aplikacji Ampio UNI.

