

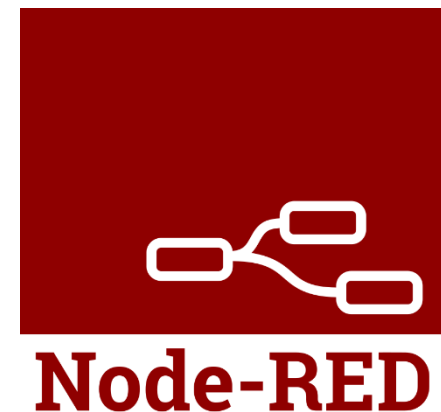
# Node-RED

## Základy internetu vecí

ÚINF PF UPJŠ

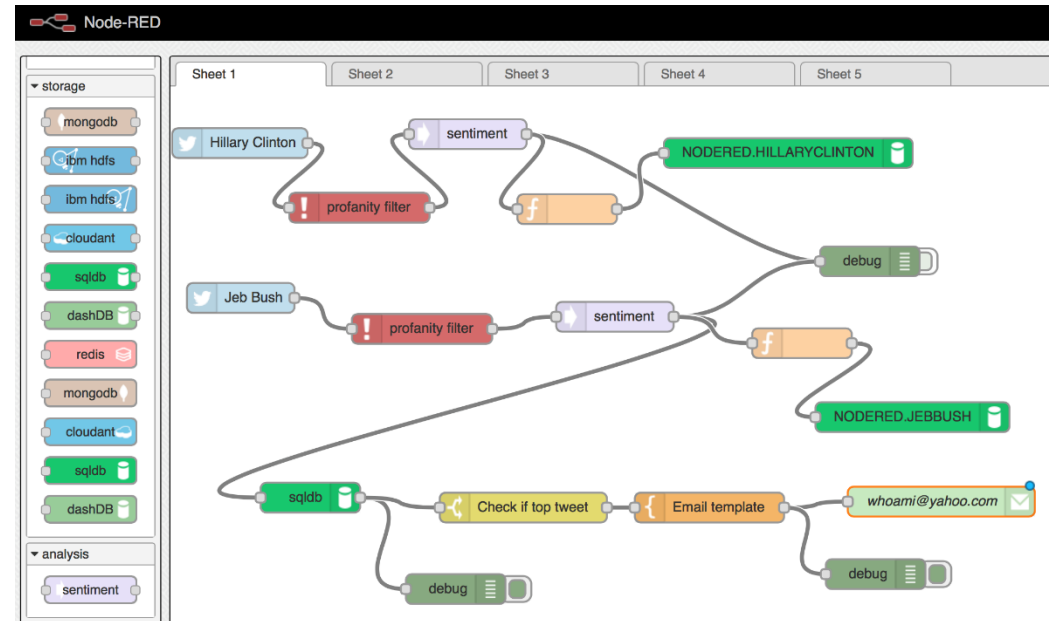
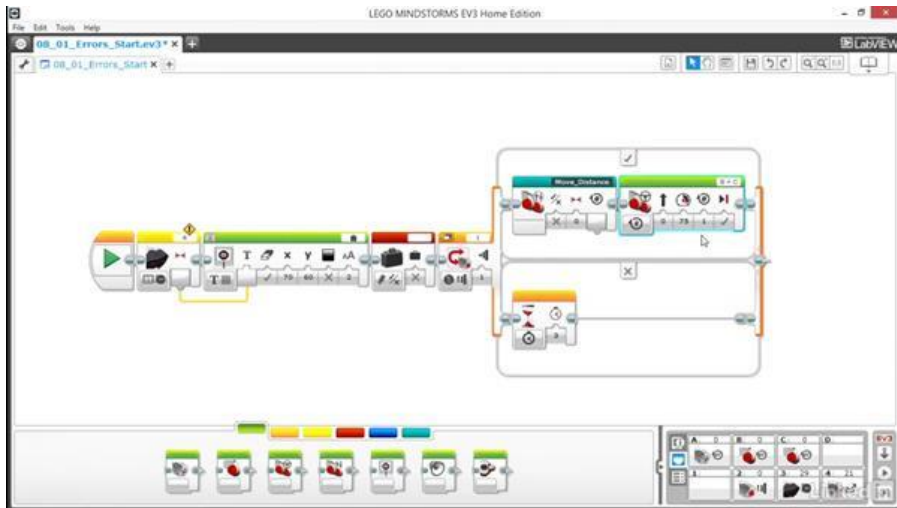
# Node-RED

- **Node-RED** is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.
- It provides a browser-based editor that makes it easy to **wire** together **flows** using the wide range of **nodes** in the palette that can be deployed to its runtime in a single-click.
- *podľa <https://nodered.org>*



# Node-RED

- Flow-based programming (J. Paul Morrison, 70-te roky)
- Vedľajší projekt v IBM (Nick O'Leary and Dave Conway-Jones, 2013)
- od 09/2013 - open-source
- Súčasť Raspbian



# Inštalácia a spustenie

- Raspbian - predinštalované
- Prerekvizita - **Node.js**
- Spustenie
  - `node-red`
  - `http://localhost:1880`
- automatické spustenie na Raspberry Pi
  - `sudo systemctl enable nodered.service`
- RPi: `node-red-start`  
(zatvorenie terminálu neukončí beh programu)

# Ukážka č.1 – formátovaný timestamp

- Uzly (Nodes):
  - inject timestamp
  - debug
  - function



# Node-RED základy

- Uzly (**Nodes**)
  - vstupné (**input**)
  - vystupné (**output**)
  - vstupno-výstupné (**processing**)
- Node-RED program – flow na základe spojených uzlov
- **Správa** - hlavná časť = msg.payload
  - vo väčšine prípadov jediná vec, s ktorou uzly pracujú -> bezstavovosť (okrem contextu)
- **Wires** – spojenia určujú ako prúdia správy
  - môžu byť viacnásobné, správa sa posiela do všetkých vetiev
- Flow import/export vo formáte JSON

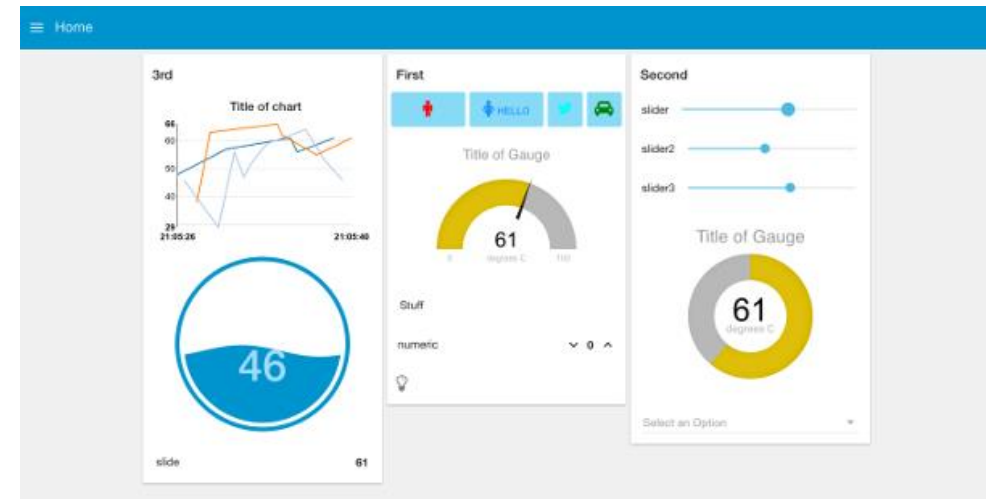
# Ukážka č.2 – filtrovanie faktúr

- Otvorené dáta – datasey na data.gov.sk
- OPEN Data mesta Prešov
  - <http://egov.presov.sk/default.aspx?NavigationState=1100:0>:
- Dodávateľské faktúry
  - JSON formát
  - Veľké zákazky (nad nejakú sumu) zapísať do súboru
  - Prepočet na inú menu

```
{
  "Interné_číslo_faktúry": "2015000317",
  "Názov_organizácie": "Mesto Prešov",
  "Dodávateľ": "Správa a zimná údržba prešovských ciest, s.r.o.",
  "Popis_účel_platby": "- za vykonané práce: zimná údržba",
  "Celková_cena": " 45 392,75",
  "Mena": "EUR",
  "Dátum_úhrady": "20.03.2015",
  "Centrálne_číslo_zmluvy": "",
  "Číslo_objednávky": "",
  "Dátum_zverejnenia": "11.03.2015",
  "IČO_-_Identifikačné_číslo_dodávateľa": "31719473 ",
  "Adresa_dodávateľa": "Pionierska 442/24, Prešov, 08005",
  "Stav_zaplatenia": "za platené",
  "Štruktúra_úradu": "V. Sekcia vnútornej správy a VČ",
  "Číslo_faktúry": "20150103",
  "Finálna_/Zálohová": "Finálna",
  "Stav_vybavenia": ""
}
```

# Ukážka č. 3 – IoT

- **MQTT**
  - test.mosquitto.org:1883
  - Sensor Node free (google play)
- **Dashboard**
  - Vizualizácia údajov zo senzorov
  - Manage palette -> Install
  - `node-red-dashboard`
- **Arduino**
  - Serial communication, Firmata, Johnny-Five
  - Arduino IDE: File > Examples > Firmata > StandardFirmata
  - `node-red-node-arduino`





# Pod kapotou Node-RED

- Flow - spojená kolekcia uzlov, ktorá si vymieňa správy
- **JavaScript objekty** popisujú **uzly** a ich konfiguráciu
- **Správy** – **JS objekty**
- **Uzly** – **JavaScript**, ktorý beží v Node-RED service + **html** popis uzlu
  
- **Context** - function uzly majú prístup k lokálnemu kontextu - indexovanie, výpočty s dátami z viacerých správ (*ukážka - počítadlo*)
- **Global context** - na zdieľanie dát medzi všetkými uzlami (napr. na odmeranie času)

# Kedy použiť Node-RED?

- *Ked' posielam dáta odniekiaľ niekam a cestou ich trochu mením*
- Výhody Node-RED
  - **Flow-based** dobre modeluje svet IoT (niečo spúšťa inú akciu)
  - **Building blocks** – dostupné uzly
- Kedy Node-RED nehodí
  - veľmi komplexný systém - neprehľadný vo vizualizácii
  - cykly
  - špecifický use case – Node-RED môže spájať jednotlivé časti

Ďakujem za pozornosť