Node-red@Druidlab: Living Lab 5/6/2024

Dataset
The json in appendix (*) contains a list of sensor data (accelerator data) of different devices collected on a certain date.

The json file has the format:

[USER_1, USER_2, …]

where, for each **i in [1..6]**

```
 USER_i is a json object
   {
     “userid”: i,     //e.g. “userid” is a unique id of a  device
     “date”: …,
     “value”: [ v1, v2,…, vn ]
   }
```

E.g. for two user ids' only:

```
[
  {
       "userid": 1,
       "date": "2022-06-29 07:39:04+00:00",
       "value": [
           0.8515625,
           0.84375,
           0.8359375
        ]
    },
  {
       "userid": 2,
        "date": "2022-06-29 07:39:05+00:00",
         "value": [
           0.8515625,
           0.84375
           ]
  }
]
```

Step 1

Select one userid K in between 1-6.
Create a flow that simulates a sensor that emits the data contained in the json object in the dataset appendix:

- Save the dataset in a json file
- Open the json
- Transform the json string into an array of objects
- Split the array into single objects (one per user)
- Extract the object with userid = K
- Extract the array of values
- Split the values and send each of them, 1 value per sec, via MQTT to the server mqtt.eclipseprojects.io on the topic /dibris/useridK

Hints:
Use the following set of nodes (see doc: https://nodered.org/docs/user-guide/nodes and palette in the Node-red editor):

- **Common**: inject (to fire the flow) and debug (to print on the console)
- **Storage:** read file to open an input stream from a file
- **Function:**
  function (to write NodeJs code), switch (to filter on certain properties)
  delay (to limit the rate at which objects can pass through the flow)
- **Parser:** json (converts a json string into a js (list of) object(s))
- **Network**: mqtt in/out (to send/receive via a broker)
- **Sequence:** split (to split arrays into an object stream)

Step 3
Create you own dashboard using the **dashboard node** (see https://flows.nodered.org/node/node-red-dashboard/), e.g. using charts.

Step 4
Add GPS coordinates to the each value extracted from the file and display the data on a map using a **location** node (see doc: https://flows.nodered.org/node/node-red-contrib-web-worldmap)

An instance of Node-red is available on the VM druidlab.dibris.unige.it:
https://druidlab.dibris.unige.it:8088/

Dataset: druidlab server at “**/home/uc/data/data.json",**

```
[
    {
        "userid": 1,
        "date": "2022-06-29 07:39:04+00:00",
        "value": [
            0.8515625,
            0.84375,
            0.8359375,
            0.7578125,
            0.73828125,
            0.71875,
            0.66015625,
            0.66796875,
            0.69140625,
            0.69140625,
            0.7109375,
            0.73046875,
            0.7421875,
            0.7734375,
            0.76953125,
            0.734375,
            0.71875,
            0.72265625,
            0.6796875,
            0.7109375,
            0.73046875,
            0.7421875,
            0.75,
            0.7578125,
            0.76953125,
            0.734375,
            0.72265625,
            0.6796875,
            0.71875,
            0.73828125,
            0.7421875,
            0.74609375,
            0.7421875
        ]
     },
    {
        "userid": 2,
        "date": "2022-06-29 07:39:05+00:00",
        "value": [
            0.8203125,
            0.7734375,
            0.76953125,
            0.734375,
            0.6875,
            0.66015625,
            0.66796875,
            0.69140625,
            0.7109375,
            0.73046875,
        0.7421875,
        0.75,
        0.8515625,
        0.84375,
        0.8359375,
        0.8203125,
        0.76953125,
        0.734375,
        0.6875,
        0.66015625,
        0.66796875,
        0.69140625,
        0.7109375,
        0.73046875,
        0.7421875,
        0.7734375,
        0.76953125,
        0.734375,
        0.71875,
        0.72265625,
        0.6796875,
        0.71875,
        0.6875,
        0.66015625,
        0.66796875,
        0.69140625,
        0.69140625,
        0.7109375,
        0.73046875,
        0.7421875,
        0.7734375,
        0.76953125,
        0.734375,
        0.71875,
        0.72265625,
        0.6796875,
        0.7109375,
        0.73046875,
        0.7421875,
        0.75,
        0.7578125,
        0.76953125,
        0.734375,
        0.71875,
        0.72265625,
        0.6796875,
        0.7109375,
        0.73828125,
        0.73828125,
        0.7421875,
        0.74609375,
        0.7421875
    ]
},
     {
    "userid": 3,
    "date": "2022-06-29 07:39:05+00:00",
    "value": [
        0.8515625,
        0.84375,
        0.8359375,
        0.8203125,
        0.7734375,
       0.76953125,
       0.734375,
       0.71875,
       0.72265625,
       0.6796875,
       0.71875,
         0.7109375,
       0.73046875,
       0.7421875,
       0.7734375,
       0.76953125,
       0.734375,
       0.71875,
       0.72265625,
       0.6796875,
       0.7109375,
       0.73046875,
       0.7421875,
       0.75,
       0.7578125,
       0.76953125,
       0.6875,
       0.66015625,
         0.7109375,
       0.73046875,
       0.7421875,
       0.7734375,
       0.76953125,
       0.734375,
       0.71875,
       0.72265625,
       0.6796875,
       0.7109375,
       0.73046875,
       0.7421875,
       0.75,
       0.7578125,
       0.76953125,
       0.66796875,
       0.69140625,
       0.7109375,
       0.73046875,
       0.7421875,
       0.75,
       0.7578125,
       0.73828125,
       0.73828125,
       0.7421875,
       0.74609375,
       0.7421875
   ]
},
    {
   "userid": 4,
   "date": "2022-06-29 07:39:05+00:00",
   "value": [
       0.8515625,
       0.84375,
       0.8359375,
       0.8203125,
       0.7734375,
       0.76953125,
       0.734375,
        0.71875,
        0.76953125,
        0.734375,
        0.71875,
        0.72265625,
        0.6796875,
        0.7109375,
        0.73046875,
        0.7421875,
        0.75,
        0.72265625,
        0.6796875,
        0.71875,
        0.6875,
        0.734375,
        0.71875,
        0.72265625,
        0.6796875,
        0.7109375,
        0.73046875,
        0.7421875,
        0.75,
        0.66015625,
        0.66796875,
        0.69140625,
        0.7109375,
        0.73046875,
        0.7421875,
        0.75,
        0.7578125,
        0.73828125,
        0.73828125,
        0.7421875,
        0.74609375,
        0.7421875
    ]
} ,
     {
    "userid": 5,
    "date": "2022-06-29 07:39:05+00:00",
    "value": [
        0.8515625,
        0.84375,
        0.8359375,
        0.8203125,
        0.7734375,
        0.76953125,
        0.734375,
        0.71875,
        0.72265625,
        0.6796875,
        0.71875,
        0.6875,
        0.734375,
        0.71875,
        0.72265625,
        0.6796875,
        0.7109375,
        0.73046875,
        0.7421875,
        0.72265625,
        0.6796875,
        0.71875,
           0.6875,
           0.66015625,
           0.66796875,
           0.69140625,
           0.7109375,
           0.6875,
           0.734375,
           0.71875,
           0.72265625,
           0.6796875,
           0.7109375,
           0.73046875,
           0.7421875,
           0.75,
           0.7578125,
           0.73828125,
           0.73828125,
            0.7421875,
           0.72265625,
           0.6796875,
           0.7421875,
           0.74609375,
           0.7421875
       ]
    } ,
    {
       "userid": 6,
       "date": "2022-06-29 07:39:05+00:00",
       "value": [
           0.8515625,
           0.84375,
           0.8359375,
           0.8203125,
           0.7734375,
           0.734375,
           0.71875,
           0.72265625,
           0.6796875,
           0.7109375,
           0.73046875,
           0.7421875,
           0.76953125,
           0.734375,
           0.71875,
           0.72265625,
           0.6796875,
           0.71875,
           0.6875,
           0.72265625,
           0.6796875,
           0.71875,
           0.6875,
           0.66015625,
           0.66796875,
           0.69140625,
           0.6796875,
           0.7109375,
           0.73046875,
           0.7421875,
           0.75,
           0.7109375,
           0.6875,
           0.734375,
           0.71875,
           0.72265625,
           0.7578125,
           0.73828125,
           0.73828125,
            0.7421875,
           0.72265625,
           0.6796875,
           0.7421875,
           0.74609375,
           0.7421875
       ]
    }
]
```