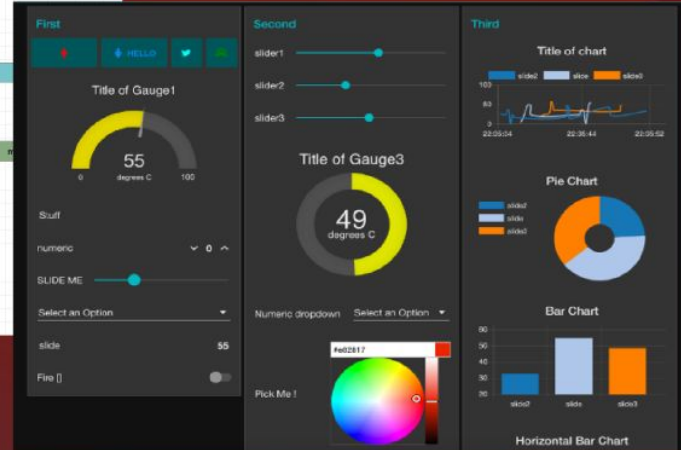
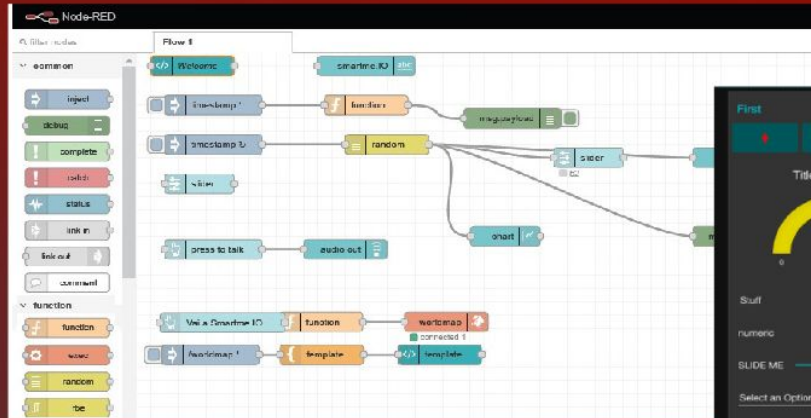
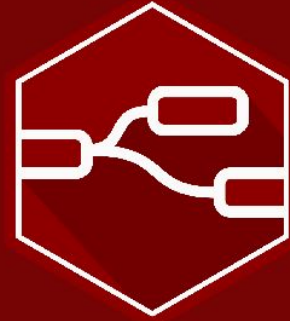


Node-RED

Il coltellino svizzero del maker



fablabroma

Ivan Tarozi

MakerSpace Santarcangelo 23/03/2024



**Node-RED nasce in IBM nel 2013
Rilasciato come open-source lo stesso anno**

Partecipa dapprima alla JS Foundation nel 2016

Nel 2019 confluisce nella OpenJS Foundation

**<https://nodered.org/about/#history>
<https://flowfuse.com/blog/2024/02/history-of-nodered/>**



Essendo basato su NodeJS può venire eseguito su una grande varietà di piattaforme, dal Cloud al RaspberryPI



Running locally

Installing Node-RED on your local computer



Raspberry Pi

Get started using our all-in-one install script for the mighty Raspberry Pi



Docker

Running Node-RED using Docker



Install from git

Building Node-RED from source. Get the very latest development code and start contributing.



BeagleBone Boards

Running Node-RED on BeagleBone boards



Android

A bit experimental, but you can run on Android devices using Termux



FlowFuse

Running a multi-tenant Node-RED solution



AWS

Get started running on Elastic Beanstalk or EC2



Microsoft Azure

Running on an Azure Virtual Machine Instance

<https://nodered.org/docs/getting-started/>



Documentation



Getting Started

Everything from first install to
deploying flows



User Guide

The definitive guide to using
Node-RED



Frequently Asked Questions

And hopefully some answers



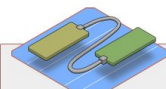
Tutorials

Examples of what you can do,
taken one step at a time



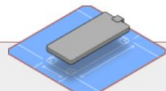
Cookbook

Recipes to help you get things
done with Node-RED



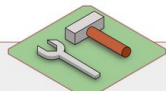
Developing Flows

Best practices for creating
clear and reusable flows



Creating Nodes

How to create nodes to
extend the Node-RED palette



Developing the core

Help to develop the core of
Node-RED



API Reference

Admin, runtime and storage
APIs

Aricchito da una buona
documentazione e da una
community attiva

Trattandosi di Software Libero e
Open Source è possibile integrare
nuove parti o contribuire allo
sviluppo del progetto

<https://nodered.org/docs/>



fablabromagna

Node-RED si utilizza da Web Browser



Avvio Node-RED da terminale

Il servizio rimane in attesa sulla porta di default 1880

```
> node-red
20 Mar 12:22:07 - [info]

Welcome to Node-RED
=====

20 Mar 12:22:07 - [info] Node-RED version: v3.1.0
20 Mar 12:22:07 - [info] Node.js version: v21.6.2
20 Mar 12:22:07 - [info] Linux 6.7.7-1-MANJARO x64 LE
20 Mar 12:22:07 - [info] Loading palette nodes
20 Mar 12:22:08 - [info] Worldmap version 4.6.2
20 Mar 12:22:08 - [info] Dashboard version 3.6.2 started at /ui
(node:9867) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use
util.
(Use 'node --trace-deprecation ...' to show where the warning was created)
20 Mar 12:22:08 - [info] Settings file : /home/ivan/.node-red/settings.js
20 Mar 12:22:08 - [info] Context store : 'default' [module=memory]
20 Mar 12:22:08 - [info] User directory : /home/ivan/.node-red
20 Mar 12:22:08 - [warn] Projects disabled : editorTheme.projects.enabled=false
20 Mar 12:22:08 - [info] Flows file : /home/ivan/.node-red/flows.json
20 Mar 12:22:08 - [info] Server now running at http://127.0.0.1:1880/
20 Mar 12:22:08 - [warn]

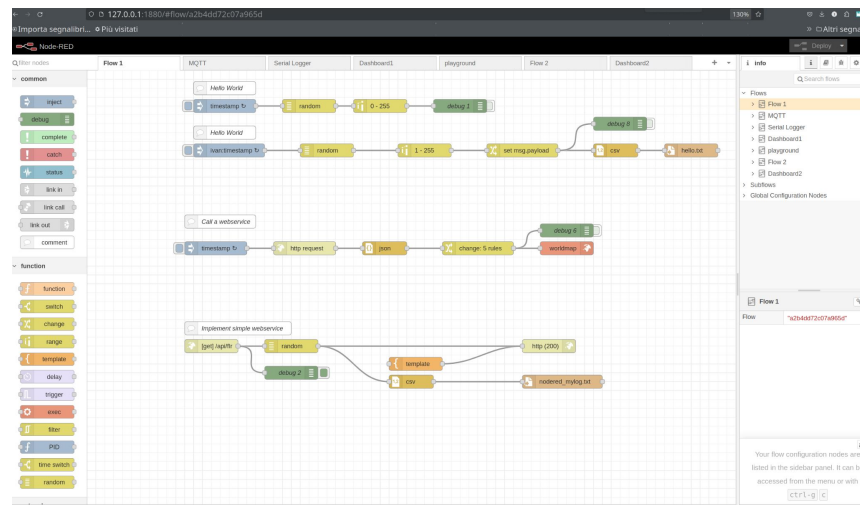
-----
Your flow credentials file is encrypted using a system-generated key.

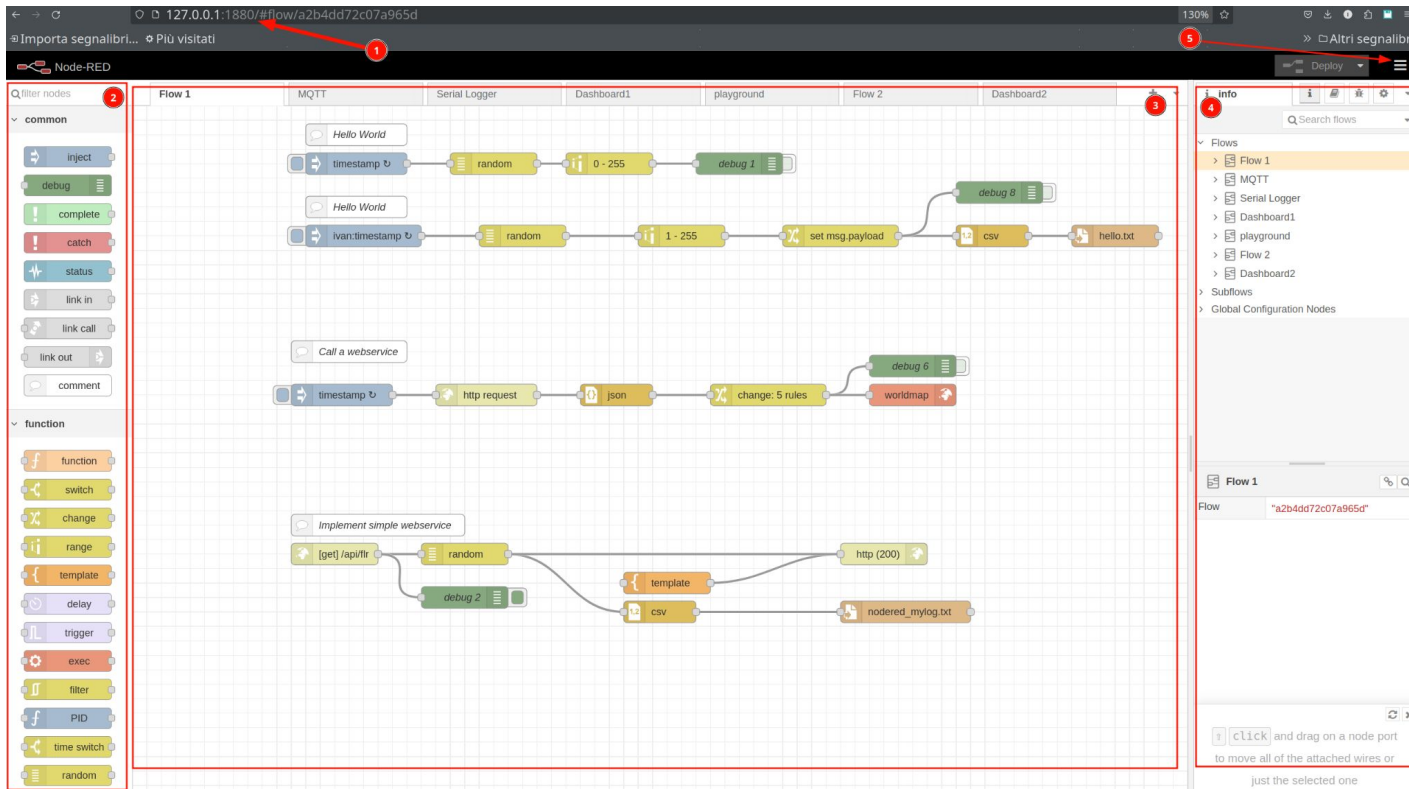
If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----

20 Mar 12:22:08 - [info] Starting flows
20 Mar 12:22:08 - [info] [ui-base:FLR Workshop] Node-RED Dashboard 2.0 (v1.3.0) started at
```

Indirizzo il web browser (127.0.0.1:1880)
per accedere all'editor



The screenshot shows the Node-RED web browser editor interface. The browser address bar shows the URL `127.0.0.1:1880/#flow/a2b4dd72c07a965d`. The interface is divided into several sections:

- 1** (Address Bar): The browser address bar showing the URL `127.0.0.1:1880/#flow/a2b4dd72c07a965d`.
- 2** (Node Palette): The left sidebar containing a search bar and a list of nodes categorized into 'common' and 'function'.
- 3** (Workspace): The central area where flows are built, showing a grid of nodes and connections. The workspace is divided into tabs for different flows: MQTT, Serial Logger, Dashboard1, playground, Flow 2, and Dashboard2.
- 4** (Info Panel): The right sidebar showing the 'info' panel for the selected flow, displaying a list of flows and the details of the selected flow.
- 5** (Menu): The top right corner of the interface, containing a menu icon and a 'Deploy' button.

- 1) IP:1880
- 2) Palette
- 3) Workspace
- 4) Sidebar
- 5) Menu



Node-RED: Hello World




```
>>> print("Hello World!")  
Hello World!  
>>>
```

```
#include <stdio.h>
```

```
int main() {  
    printf("Hello World\n");  
    return 0;  
}
```

```
console.log('Hello,  
world!');
```

 Node-RED

Q filter nodes


playground

Flow 1


MQTT

▼ common


 inject

 debug

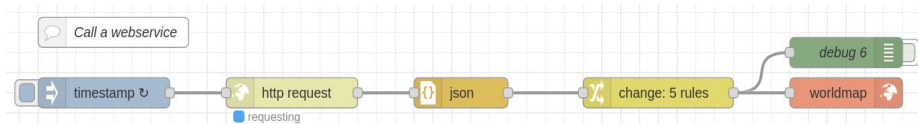
 complete

 catch

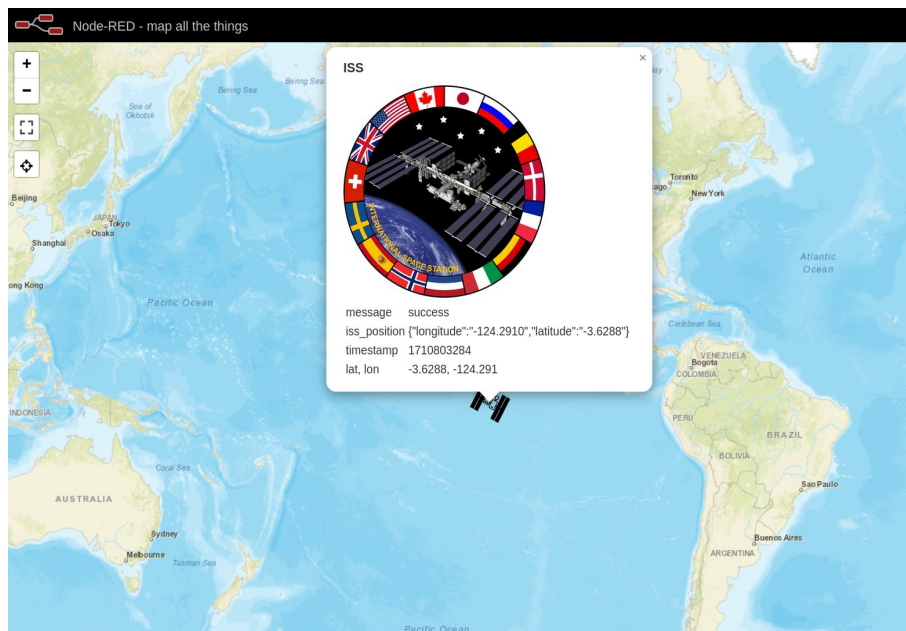
 status

 Hello World

 debug HelloWorld

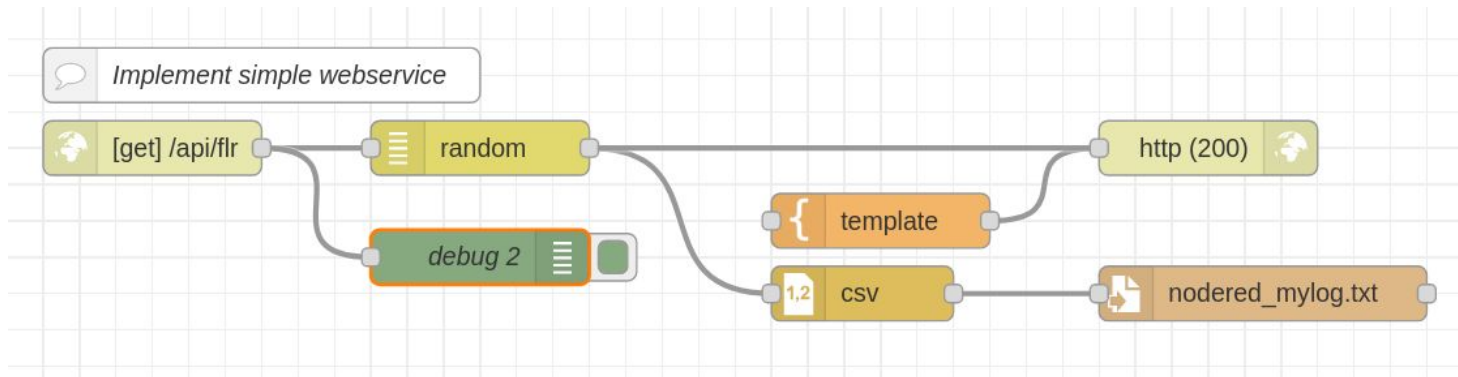


Con pochi e semplici nodi è possibile interrogare webservice e processare il JSON restituito.

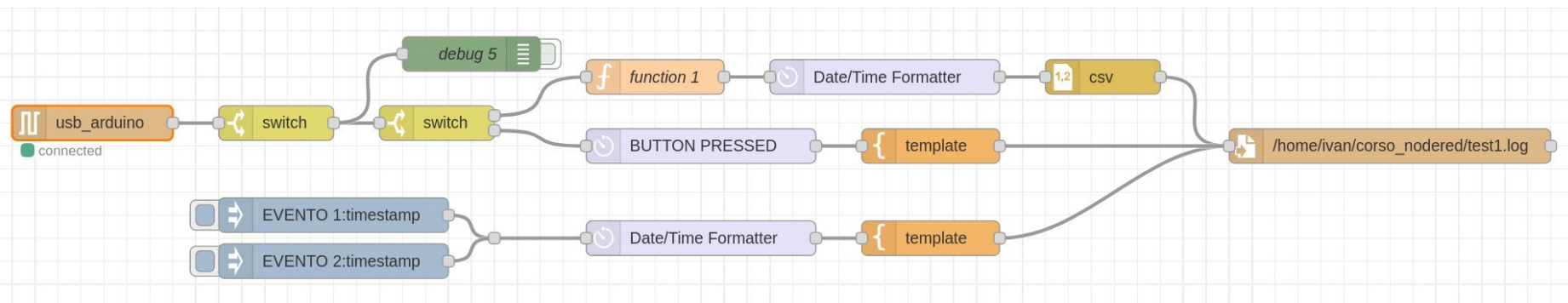


Come esempio utilizziamo l'API pubblica che fornisce in tempo reale la posizione della Stazione Spaziale Internazionale

api.open-notify.org/iss-now.json



In modo altrettanto semplice è possibile implementare una API REST oppure servire pagine HTML con dati provenienti dal Flow ed elaborati da Node-RED



Utilizzando i nodi di lettura della porta seriale è possibile implementare un debug di un programma Arduino o similare, che utilizzi le classiche funzioni `Serial.print()`

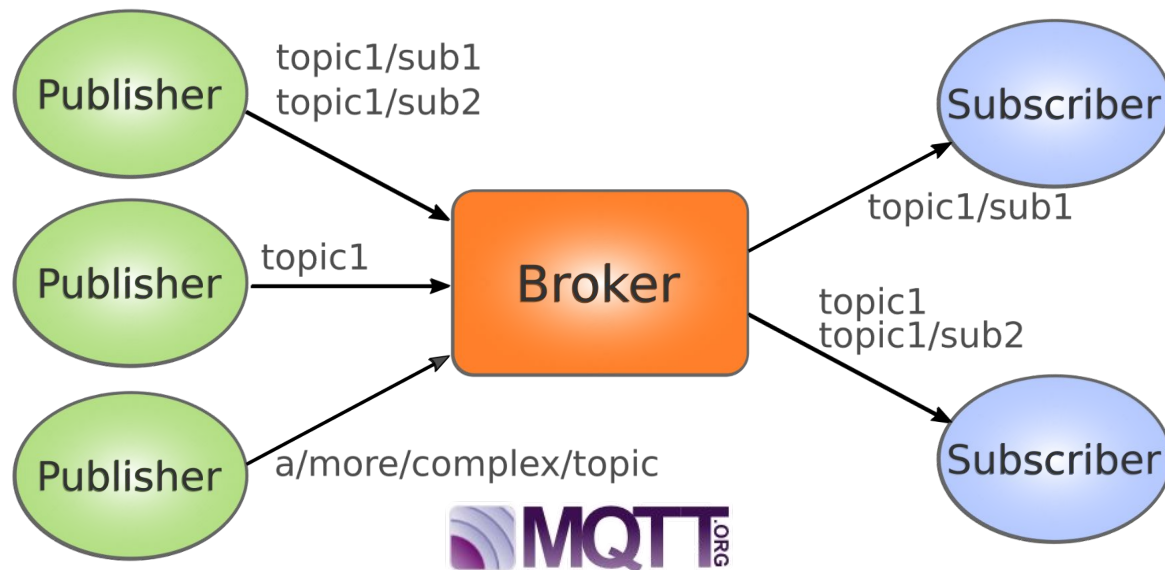
Rispetto al “monitor seriale” dell’IDE è però possibile aggiungere filtri, log di eventi custom e salvare il tutto su file CSV o database

Lo stesso concetto può essere applicato per implementare datalogger o per interfacciarsi con moduli che utilizzano la seriale e i comandi AT (ad esempio moduli **LoRaWAN** o **GPS**)



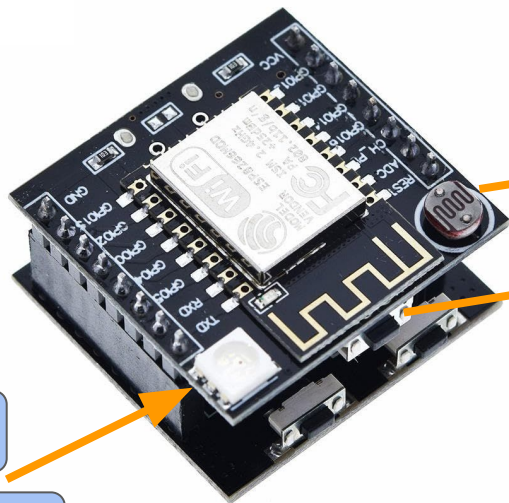
I *publisher* inviano i dati al *broker*, utilizzando come riferimento della variabile un *topic* gerarchico

I *subscriber* comunicano al *broker* interesse per uno o più *topic*, e ricevono da esso notifica quando il dato relativo viene aggiornato dal *publisher*





WITTY ESP8266



flr/ws20240316/witty1/led_onoff

flr/ws20240316/witty1/led_rgb

Subscribe

flr/ws20240316/witty1/light

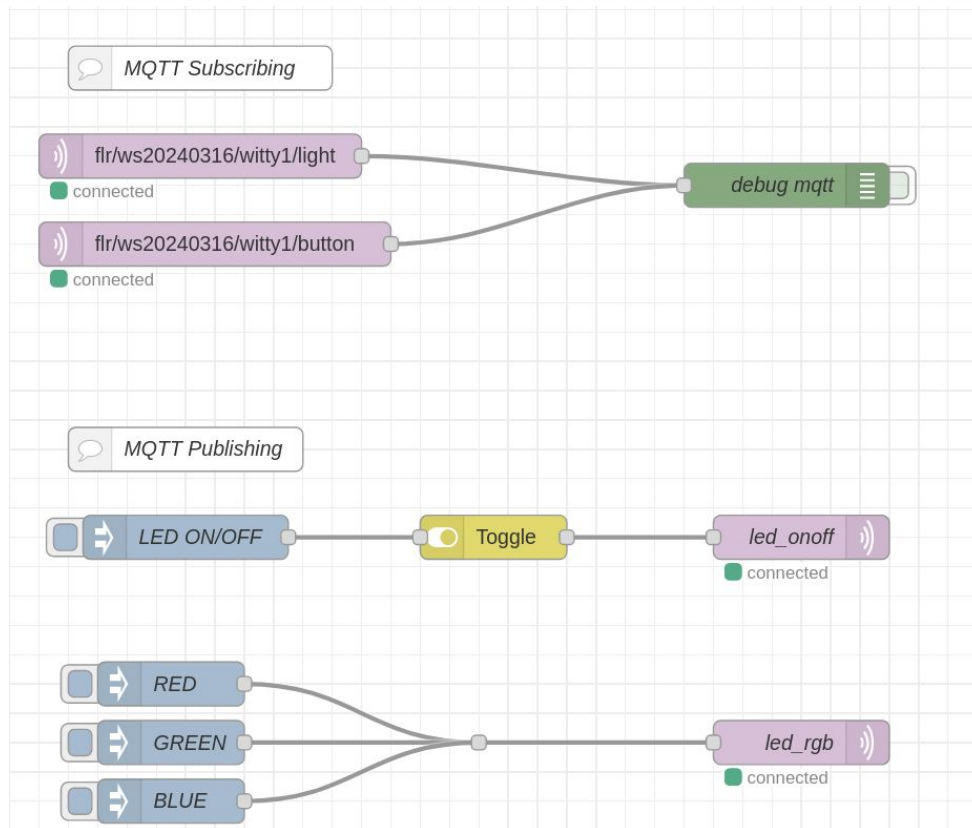
flr/ws20240316/witty1/button

Publish

MQTT Broker:
test.mosquitto.org



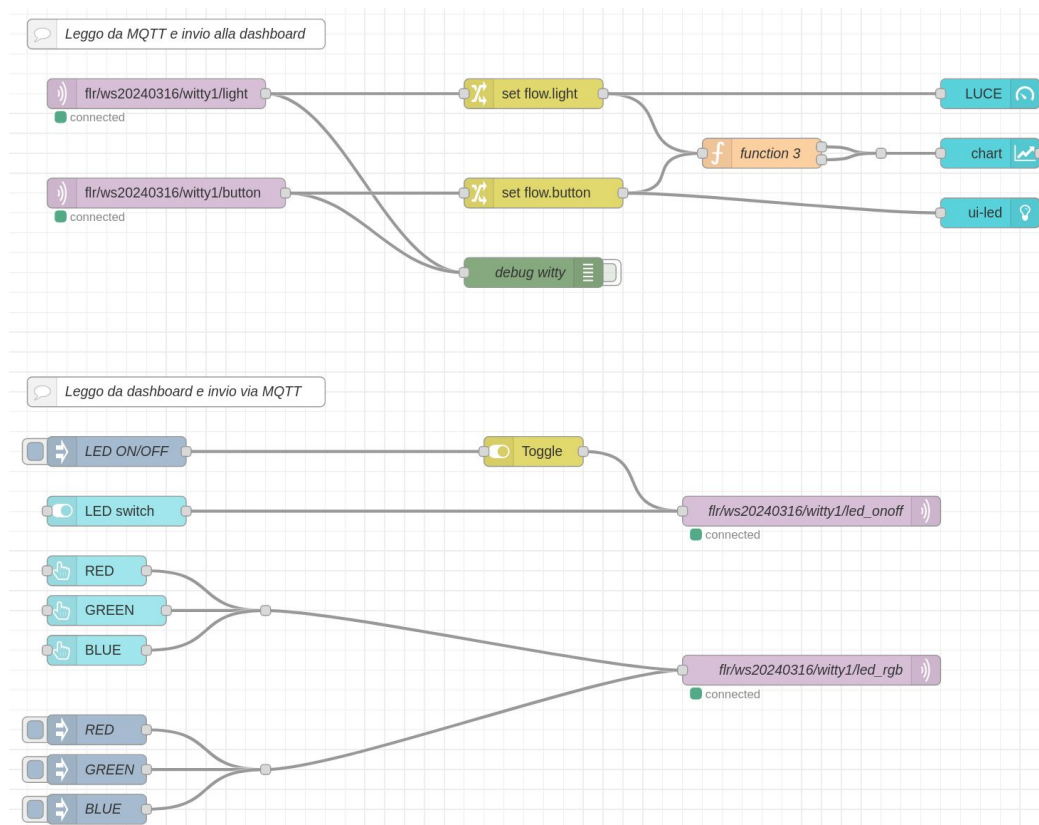
WITTY ESP8266





WITTY ESP8266

Stesso esempio con
l'aggiunta di una dashboard
per visualizzare i dati e
inviare i comandi





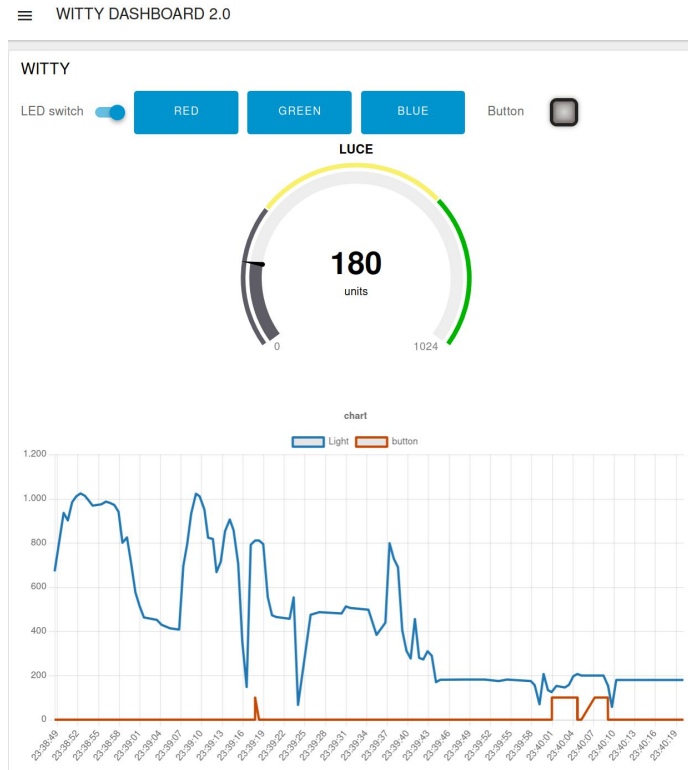
Node-RED Dashboard

Oltre alla “vecchia” dashboard è ora presente una nuova **Dashboard 2.0** più ricca di funzionalità e maggiormente personalizzabile

<https://dashboard.flowfuse.com/>

Basata su **Vue.js v3.0** e sulla [Vuetify Component Library](#) (Material Design framework for Vue.js)

https://github.com/fablabromagna-org/node-red-playground/blob/master/flows/witty_dashboard2.json





LoRaWAN - The Thing Network

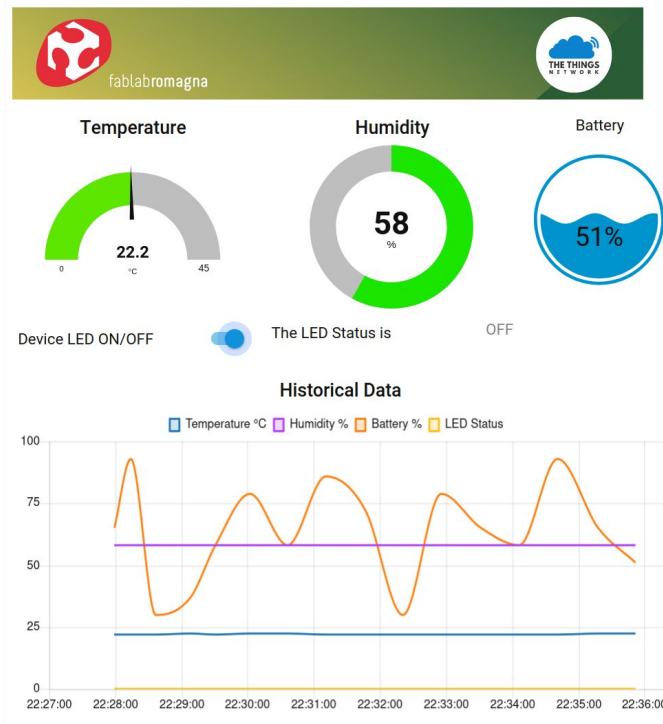
Abbiamo utilizzato Node-RED per implementare una dashboard durante la sperimentazione su LoRaWAN utilizzando il Network Server di TTN

<https://www.thethingsnetwork.org/>

TTN espone i dati ricevuti tramite una serie di protocolli tra cui MQTT

<https://github.com/fablabromagna-org/FLR-LoRaWAN>

LoRaWAN Sensor FLR-IT1 - The Things Stack MQTT Integration





Di seguito una serie di link utili per approfondire lo studio di Node-RED:

- Node-RED home page: <https://nodered.org/>
- Node-RED Library: <https://flows.nodered.org/>
- Il blog di FlowFuse: <https://flowfuse.com/blog/>
- Video corso in italiano: <https://bit.ly/nodered-corso>



Potete trovare le slide e i materiali della presentazioni sul nostro repository GitHub:

<https://github.com/fablabromagna-org/node-red-playground>



Questa opera di Fablab Romagna A.P.S è concessa in licenza sotto la
Licenza Creative Commons

Attribuzione - Condividi allo stesso modo 4.0

<https://creativecommons.org/licenses/by-sa/4.0/>

