

Node-RED installation and MariaDB persistence

Download & Install Node.js

Node.js is a library specialized in network and written in javascript.

For debian you must use this link : <https://nodered.org/docs/getting-started/raspberrypi>

It is good for raspberry and debian linux.

After you have access to nodeRED with <http://<hostname>:1880>.

For me i have a rewrite for access from a normal url.

In nodeRED you can add the plugins into the palette :

You must add these plug-ins to the project with manage palette>install :


1. @studiobox/node-red-contrib-ui-widget-humidity-tree
2. @studiobox/node-red-contrib-ui-widget-thermometer
3. node-red 3.1.3
4. node-red-contrib-config
5. node-red-contrib-ui-artless-gauge
6. node-red-contrib-ui-spinner
7. node-red-contrib-ui-time-scheduler
8. node-red-dashboard 3.6.2
9. node-red-node-mysql 2.0.0
10. node-red-node-ui-lineargauge 0.3.7

For saving globals variables of the project after each deploiement . You must modify your file settings.js in /<user nodered>/node-red/settings.js and add

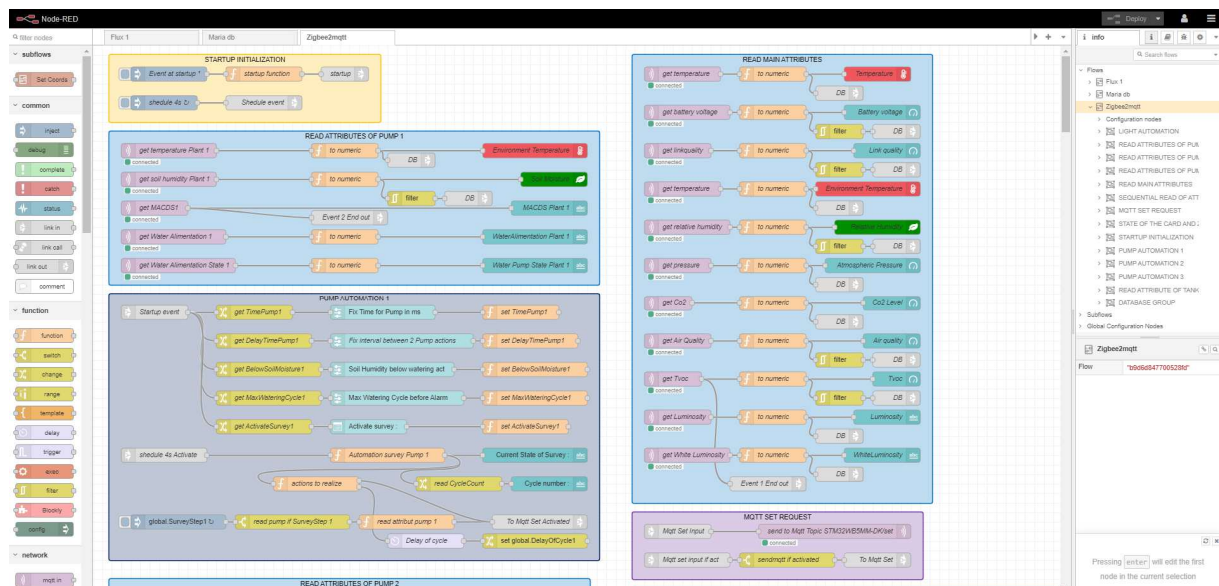
```
contextStorage: {  
  default: {  
    module:"localfilesystem"  
  },  
},
```

You must restart with : service nodered restart

From the file

 zigbee2mqtt nodered flows

You have access to the imported flow :

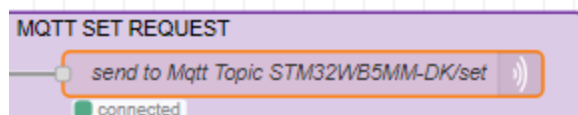


You must change the zigbee node friend name (fixed in Zigbee2mqtt) in all mqtt requests if it is not the same ... here **STM32WB5MM-DK**

zigbee2mqtt/STM32WB5MM-DK/set

After you must configure address, username, password of your mqtt server.

Click on :



You select the mqtt server

Edit mqtt in node

Delete
Cancel
Done

Properties

Server
Mqtt vps1

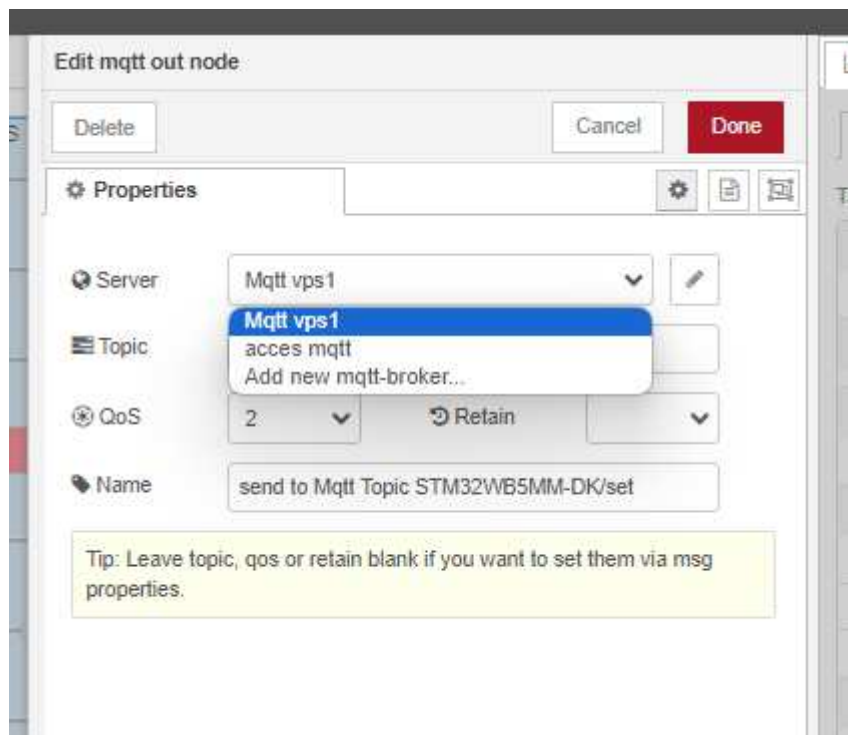
Action
Subscribe to single topic

Topic
zigbee2mqtt/STM32WB5MM-DK/temperature

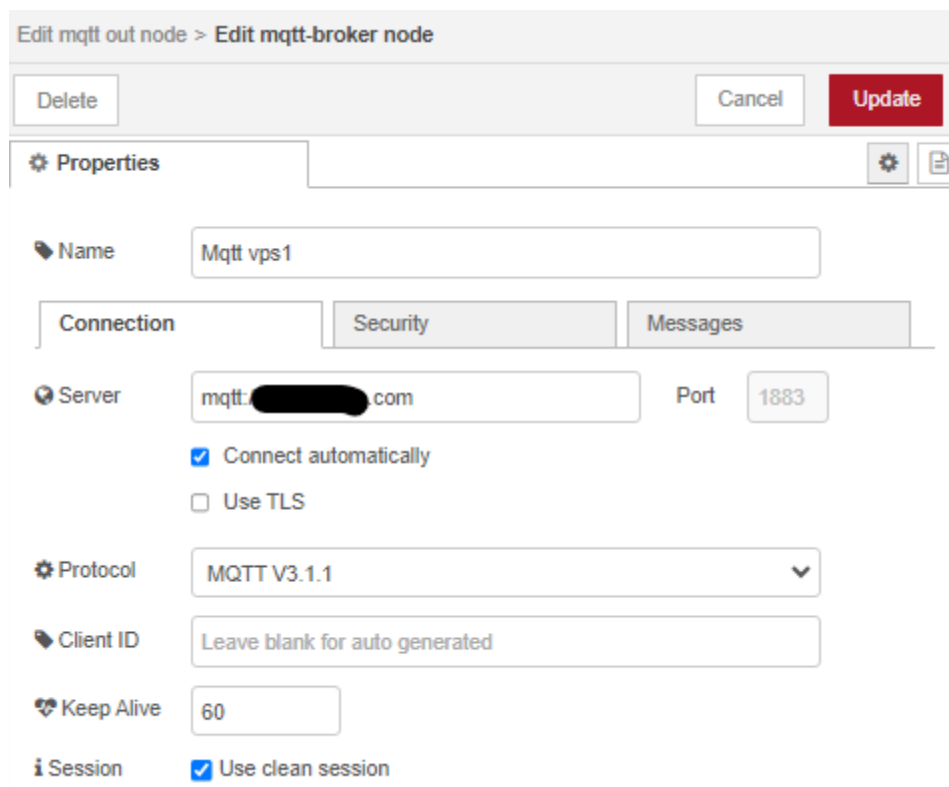
QoS
2

Output
a String

Name
get temperature



You click on pen to the right :



You can fix your mqtt server address and in security name and password.

Edit mqtt out node > Edit mqtt-broker node

Delete Cancel Update

Properties

Name Mqtt vps1

Connection Security Messages

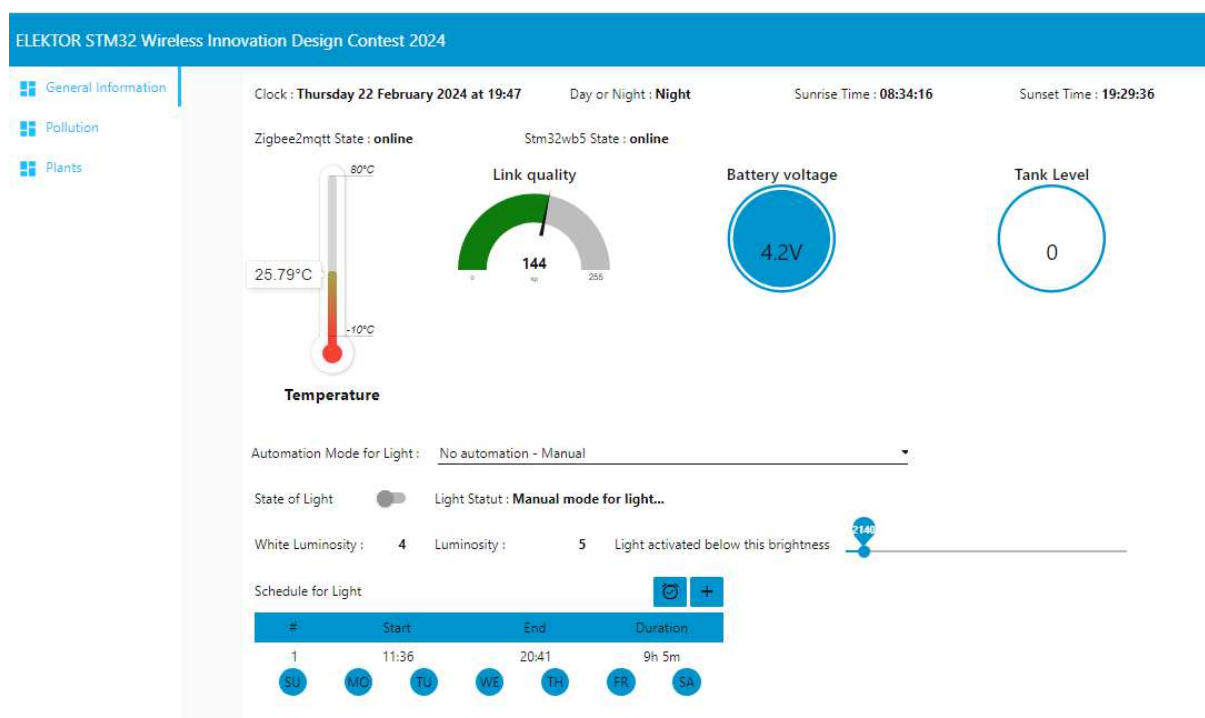
Username

Password

For each modification, click on :

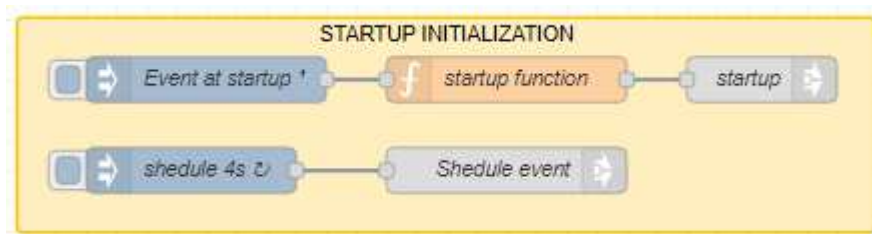


After deploy you have access to <http://localhost:1880/ui>



Parts of the project

Startup Initialization



run one time at the begin.



run every 4 seconds for scheduling the different threads.



this function initialize all the globals variables and data for day and night of your place.

```

global.set("Latitude", 42.6887);
global.set("Longitude", 2.8948);
global.set("Tz", 3600);
global.set("timezone", "Europe/Paris");
global.set("language", "en-GB");

```



anchor for send event at startup



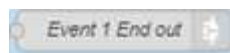
anchor for send event every 4 seconds.

Read main Attributes



Each of this type send an event when a new mqtt topic come.

The data are placed in global variable or/and to a graphical dashboard widget.



This event start the read of next attributes for no stress of zigbee.

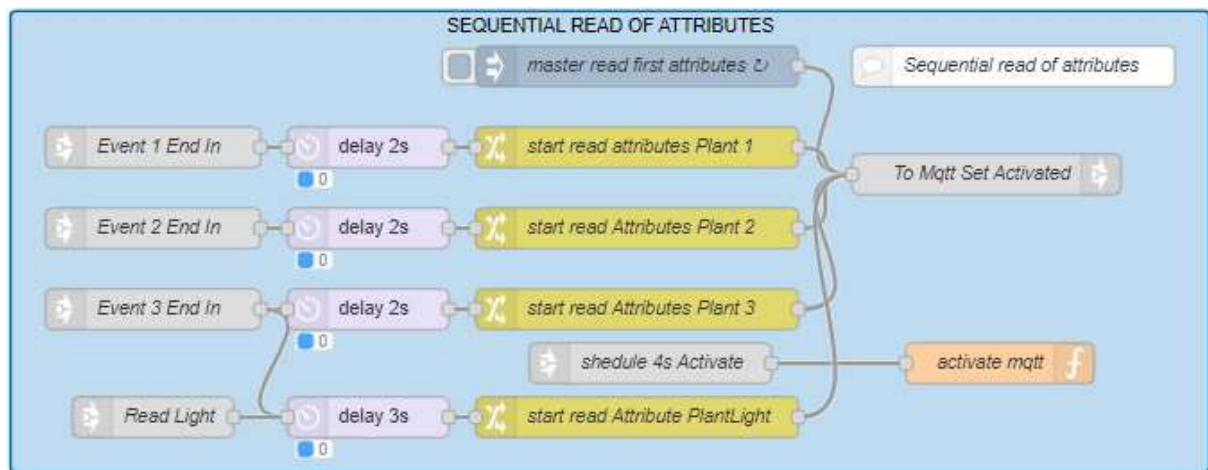
With stm32wb5 it is not possible to read more of 8 attributes at each request. (limitation ?)

MQTT SET REQUEST



this group allows you to transmit requests to read or write attributes. Two possibilities, directly or with verification of the presence of the stm32 card and active zigbee2mqtt.

SEQUENTIAL READ OF ATTRIBUTES



This group allows you to read sequentially all the attributes of the card.

For example :



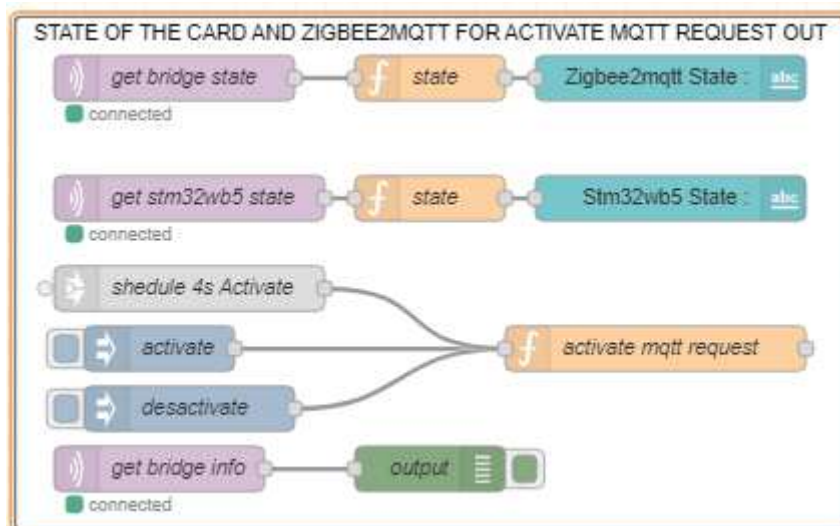
Edit JSON
Visual editor

```

1  [
2    "read": {
3      "attributes": [
4        "Luminosity",
5        "WhiteLuminosity",
6        "PlantTemperature1",
7        "PlantSoilMoisture1",
8        "PlantMACDS1",
9        "PlantWaterAlimentation1",
10       "PlantWaterAlimentationState1"
11      ],
12      "cluster": "msTemperatureMeasurement",
13      "options": {}
14    }
15  ]

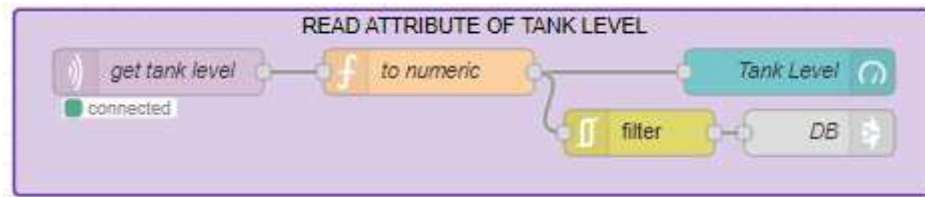
```

STATE OF THE CARD AND ZIGBEE2MQTT FOR ACTIVATE MQTT REQUEST OUT



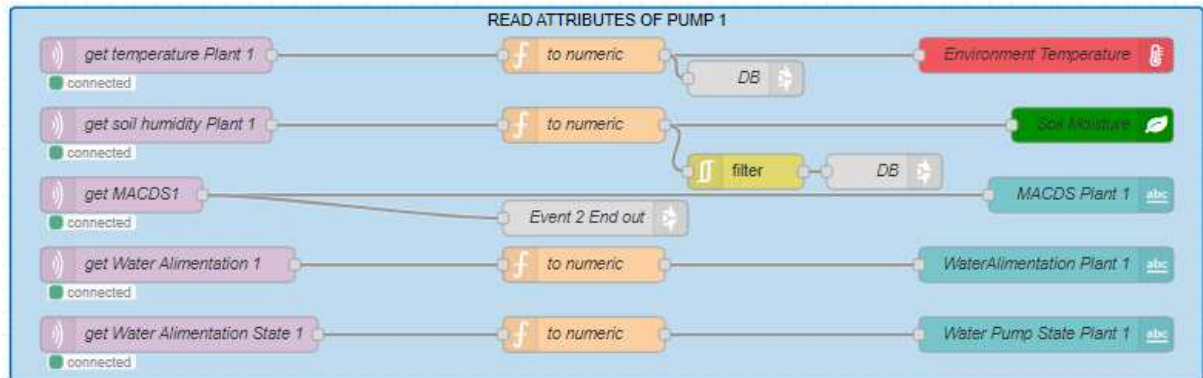
This group place the state of card and zigbee2mqtt in dashboard widget. Activate or not the mqtt requests out.

READ ATTRIBUTE OF TANK LEVEL



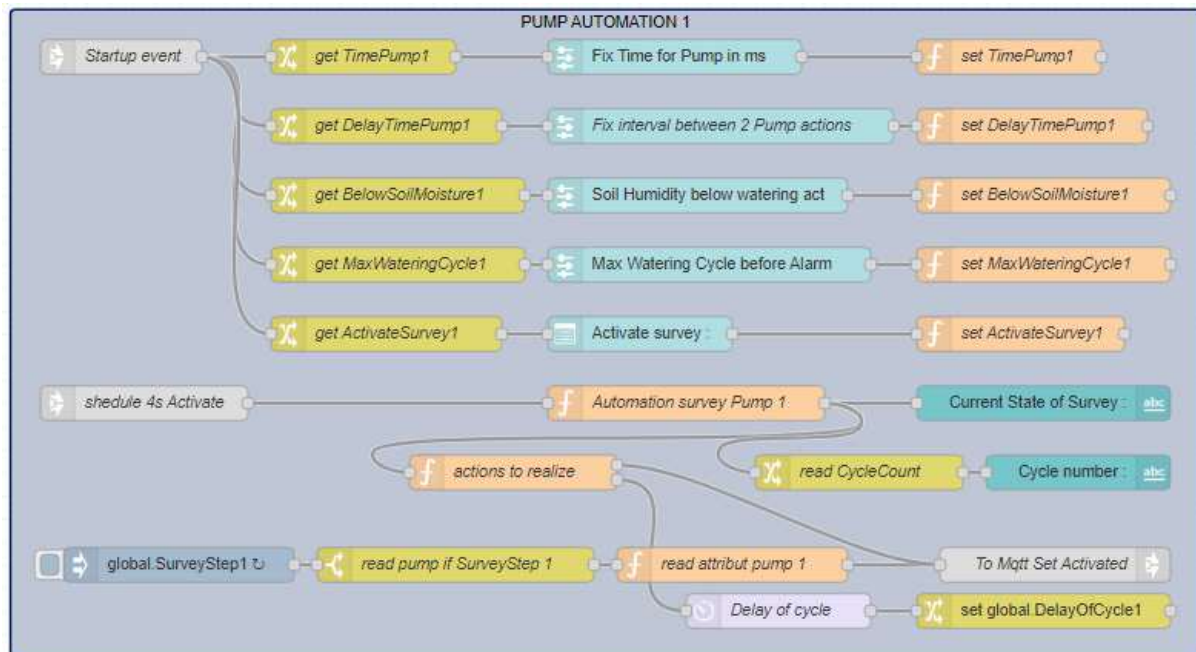
This group display the tanl level in dashboard and memorize the state.

READ ATTRIBUTES OF PUMP 1,2 and 3



Three similar groups for reading attributes of each plant part. Soil Moisture, Water Alimentation an Water alimentation state are stored in global variables. The values are display in dashboard.

PUMP AUTOMATION 1,2 and 3



Three similar group for automation of watering.

The first part allows the entry of configuration information in dashboard.



The logic is to initialize the widget with a global variable and store the new user value in this same variable.

The main function is :



```

msg.payload="";
msg.topic="";
msg.delay=0;
msg.wa="";
// verify if mqtt and stm32 board actives
if (global.get("StateSendMqtt")==1)
{
    switch(global.get("ActivateSurvey1"))
    {
        // no survey
        case 0:
            global.set("SurveyStep1",0);
            global.set("CycleCount1",0);
            msg.CycleCount = global.get("CycleCount1");
            msg.topic = "No survey activity...";
            break;
        // Automation of survey
        case 1:
            if ( (global.get("TankLevel")==0) || (global.get("MaxWateringCycle1")==global.get("CycleCount1")) )
            {
                msg.topic = "Warning : Water tank level to Low !";
                if (global.get("MaxWateringCycle1") == global.get("CycleCount1"))
                    msg.topic = "Warning : Watering stopped because Too many cycle !";
            }
            else
            {
                switch(global.get("SurveyStep1"))
                {
                    // compare soil moisture and pass to step 1
                    case 0:
                        if (global.get("SoilMoisture1") < global.get("BelowSoilMoisture1"))
                        {
                            // mqtt msg for setting action time of pump 1
                            msg.wa = {
                                "write":
                                {
                                    "cluster": "msTemperatureMeasurement",
                                    "options": {},
                                    "payload": {
                                        "PlantWaterAlimentation1": global.get("TimePump1")
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
};

```

```

        global.set("SurveyStep1",1);
        msg.topic = "Start pump...";
    }
    else
    {
        global.set("CycleCount1",0);
        msg.topic = "Survey Soil Moisture...";
    }
    break;
    // wait for pump to 0 ms
    case 1:
        if (global.get("WaterAlimentation1")==0)
        {
            global.set("SurveyStep1",2);
            msg.topic = "Start Delay for next Watering...";
            msg.delay = global.get("DelayTimePump1")*60;
            global.set("DelayOfCycle1",0);
        }
        else
        {
            msg.topic = "Pump Running...";
        }
        break;
        // wait for delay between watering act and return to 0
        // increment step and alarm if problem

    case 2:
        if (global.get("DelayOfCycle1")==1)
        {
            global.set("SurveyStep1", 0);
            global.set("CycleCount1", global.get("CycleCount1")+1);
            msg.topic = "Survey Soil Moisture...";
        }
        else
            msg.topic = "Start Delay for next Watering...";
        break;
    }
}
break;
default:
    break;
}
}
else
{
    msg.topic = "Wait for Zigbee2mqtt or stm32 activity...";
}
return msg;

```

This program if a user activate the survey with the dropdown box.

Verify if the tank level is Hight and display a message if it is too low.

Manage 3 steps by loop:

Step 0 : verify if soil moisture below the fixed set point.

If no, stay in step 0

If yes start the pump with the time fixed by the user, after pass to step 1.

Step 1 : If WaterAlimentation attribute is 0, the pump is now stopped and a delay of wait start with fixed value in minutes. (time for soil moisture react).

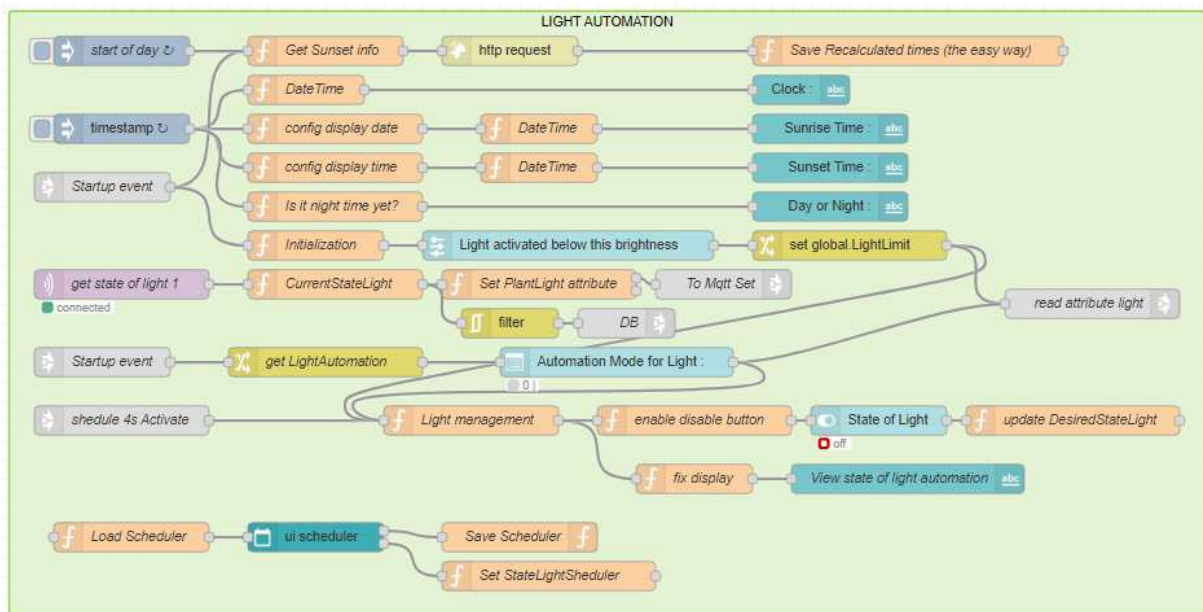
Pass to step 2.

If water alimentation attribute is 1, display « pump is running »

Step 2 : Wait end of the delay. At the end increment the counter of loop and pass to step 0 for a next loop

If the maximum of loop fixed arrived.

LIGHT AUTOMATION

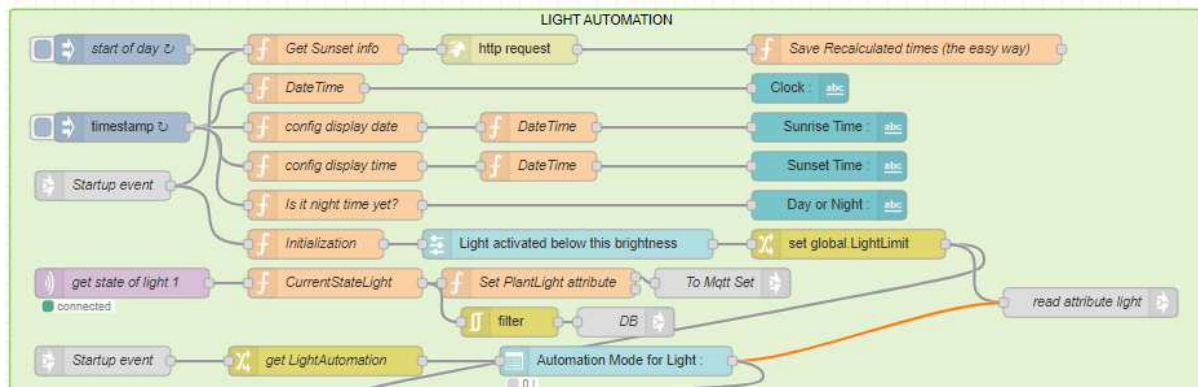


This group manage the light automation.

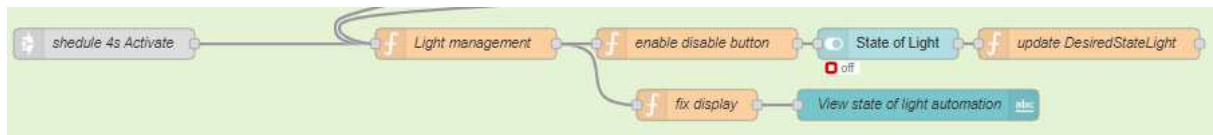
The fonctionnalités with the dropdown box in dashboard :

- Activate manually the Light.
- Activate the Light when the Brightness is too low.
- Activate the Light with a Sheduler.
- Activate the Light with a Sheduler and when the Brightness is too low.
- Activate the Light with if the Brightness is too low and if it's Day. (Sunrise and Sunset gets from website every day with API http)

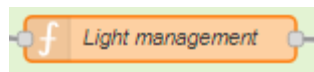
This first Part initialize the widgets, sunset,sunrise,day and night. Fix the state of light.



This part manage the automation



The program of



```
msg.payload="";
msg.topic="";
// verify if mqtt and stm32 board actives
if (global.get("StateSendMqtt")==1)
{
    switch(global.get("LightAutomation"))
    {
        // manual
        case 0:
            msg.topic = "Manual mode for light...";
            break;
        // Light On if Luminosity to low
        case 1:
            if (global.get("LightLimit") > global.get("Luminosity")) {
                msg.topic = "Light On because Brightness is low...";
                global.set("DesiredStateLight", 1);
            }
            else
            {
                msg.topic = "Light Off because Brightness is hight...";
                global.set("DesiredStateLight", 0);
            }
            break;
        // use sheduler for light on
        case 2:
            if (global.get("StateLightSheduler") == 1) {
                msg.topic = "Light On by sheduler...";
                global.set("DesiredStateLight", 1);
            }
            else {
```

```

    msg.topic = "Light Off by sheduler...";
    global.set("DesiredStateLight", 0);
}
break;
// use sheduler and level
case 3:
    if (global.get("StateLightSheduler") == 1) {
        if (global.get("LightLimit") > global.get("Luminosity")) {
            msg.topic = "Light On by sheduler and Brightness is low...";
            global.set("DesiredStateLight", 1);
        }
        else {
            msg.topic = "Light Off by sheduler and Brightness is hight...";
            global.set("DesiredStateLight", 0);
        }
    }
    else {
        if (global.get("DesiredStateLight") != 0) {
            msg.topic = "Light Off by sheduler...";
            msg.payload = 0;
        }
    }
    break;
// Light On if Luminosity to low and it's day
case 4:
    if (global.get("DayOrNight") == "Day")
    {
        if (global.get("LightLimit") > global.get("Luminosity")) {
            msg.topic = "Light On because Brightness is low and it's Day...";
            global.set("DesiredStateLight", 1);
        }
        else {
            msg.topic = "Light Off because Brightness is hight... and it's Day";
            global.set("DesiredStateLight", 0);
        }
    }
    else
    {
        msg.topic = "Light Off because it's Night";
        global.set("DesiredStateLight", 0);
    }
    break;
default:
    break;
}
}
else

```

```

{
  msg.topic = "Zigbee2mqtt or stm32 not active...";
}
msg.payload = global.get("DesiredStateLight");
return msg;

```

There are 5 operating modes.

Mode 0 : Manual mode, you can Light On/off with the button in dashboard.

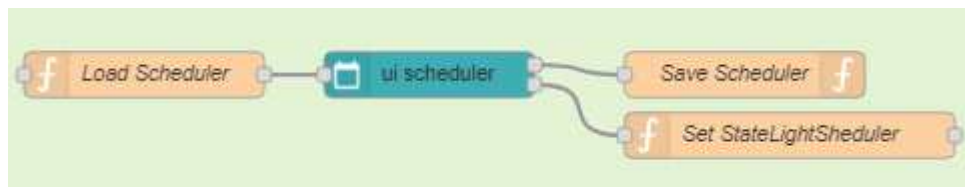
Mode 1 : The light is On if the ambient brightness is to low.

Mode 2 : The Sheduler in Dashboard fix the state of the light.

Mode 3 : The sheduler in Dashboard fix the state of the light and the light is On if the ambient Brithness i slow.

Mode 4 : If it is Day and the light is on if the ambient Brithness i slow.

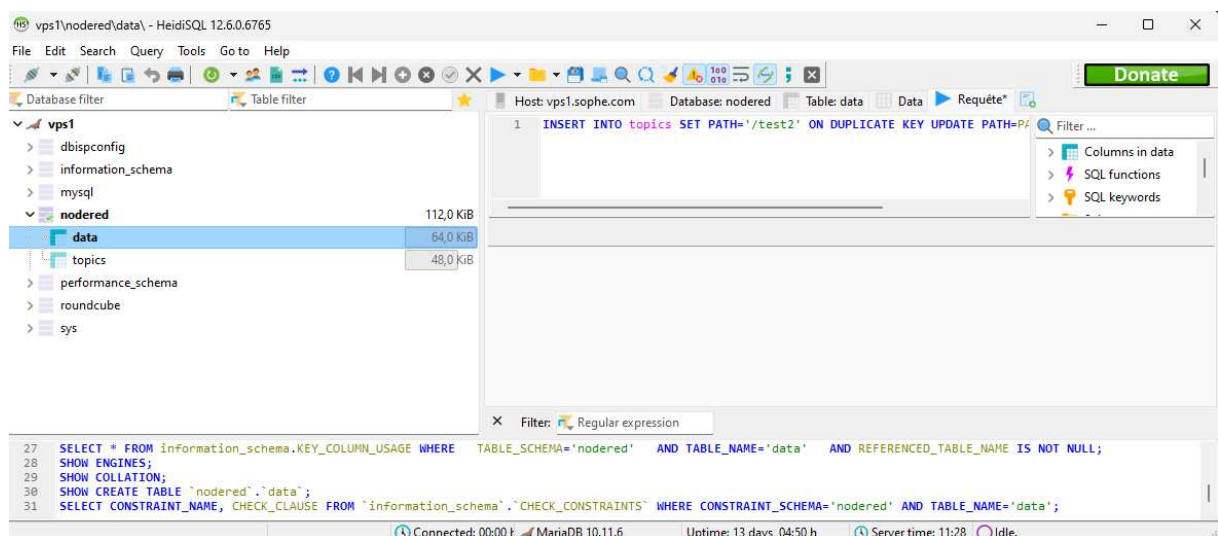
Part for UI Sheduler



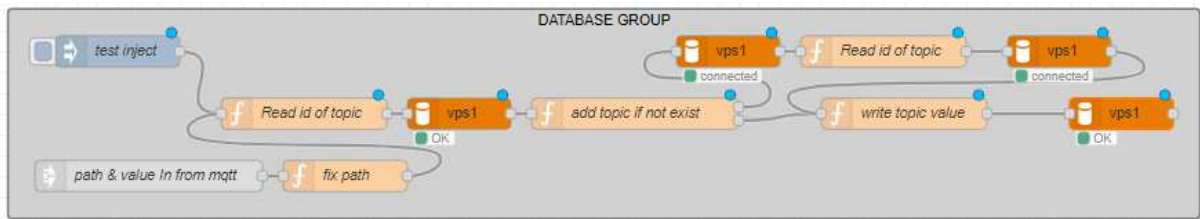
DATABASE GROUP

The next step is to report the attributes values to a database of type Mariadb for persistence.

For working on windows with a good interface to Mysql or Mariadb, you can use this very good application **heidisql** : You can download here : <https://www.heidisql.com/>



This Group manage the persistence of attributes in database



You must configure the centralised Mariadb access.



Click on

Edit mysql node

Delete Cancel Done

⚙️ Properties

📌 Name

🗄️ Database

Click on Pen on the right

Edit mysql node > Edit MySQLdatabase node

Delete Cancel Update

⚙️ Properties

📌 Name

🌐 Host

🔌 Port

👤 User

🔒 Password

🗄️ Database

🕒 Timezone

🗄️ Charset

Tip: The timezone should be specified as ±hh:mm or leave blank for "local".

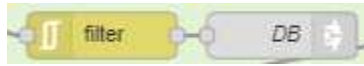
You can enter the informations about your own database.

In first time, the path topics must be created if not exist and after the value is added.

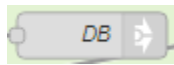


add more complex path for the value.

It is necessary to send to DB link an event with **msg.value** wich contain the **float numeric value** to store and a **msg.path** path name for the value.



With this filter, the value is memorized and the database is used only if the new value has changed since the last one.



With no filter each value is stored.

The database use two tables for reducing the size of data storage :

- **Topics table :**

Colonnes : + Ajouter ✕ Supprimer ▲ Monter ▼ Descendre									
#	Nom	Type de données	Taille/Ensem...	Non si...	NULL a...	ZERO...	Par défaut	Commentaire	Collation
1	IdTopics	INT	10	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	AUTO_INCREMENT		
2	Path	VARCHAR	255	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL		utf8mb3_general_ci

- **Data table :**

Colonnes : + Ajouter ✕ Supprimer ▲ Monter ▼ Descendre									
#	Nom	Type de données	Taille/Ensem...	Non si...	NULL a...	ZERO...	Par défaut	Commentaire	Collation
1	idtopics	INT	10	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL		
2	horodatage	TIMESTAMP	3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	current_timestam...		
3	value	DOUBLE		<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	NULL		

Sql source for create tables

```
CREATE TABLE IF NOT EXISTS `data` (
  `idtopics` int(10) unsigned DEFAULT NULL,
  `horodatage` timestamp(3) NULL DEFAULT current_timestamp(3) ON UPDATE current_timestamp(3),
  `value` double DEFAULT NULL,
  KEY `Khorodatage` (`idtopics`,`horodatage`) USING BTREE,
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3_general_ci;

CREATE TABLE IF NOT EXISTS `topics` (
  `IdTopics` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `Path` varchar(255) DEFAULT NULL,
  UNIQUE KEY `keyPath` (`Path`) USING BTREE,
  KEY `FK_topics_data` (`IdTopics`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=utf8mb3 COLLATE=utf8mb3_general_ci;
```

Now you have a persistence on selected attributes in a database.

1	4	2024-02-23 17:52:33.737	24,34
2	4	2024-02-23 17:52:45.742	24,35
3	4	2024-02-23 17:52:57.746	24,36
4	4	2024-02-23 17:53:13.960	24,36
5	6	2024-02-23 17:53:13.961	20,57
6	7	2024-02-23 17:53:13.961	977
7	11	2024-02-23 17:53:13.961	422
8	5	2024-02-23 17:53:13.962	228
9	12	2024-02-23 17:53:13.962	34
10	8	2024-02-23 17:53:13.962	4,202
11	14	2024-02-23 17:53:16.040	74
12	16	2024-02-23 17:53:16.040	19,12
13	5	2024-02-23 17:53:16.040	196
14	13	2024-02-23 17:53:16.041	42
15	17	2024-02-23 17:53:18.111	19,37
16	5	2024-02-23 17:53:20.188	216
17	4	2024-02-23 17:53:33.756	24,39
18	4	2024-02-23 17:53:57.926	24,4
19	4	2024-02-23 17:54:09.928	24,41
20	4	2024-02-23 17:54:13.959	24,41
21	6	2024-02-23 17:54:13.960	20,63
22	7	2024-02-23 17:54:13.960	977
23	11	2024-02-23 17:54:13.960	400

nodereid.topics: 19 ligne(s) au total (exact)

#	IdTopics	Path
1	4	/stm32wb5mmg/temperature
2	5	/stm32wb5mmg/LinkQuality
3	6	/stm32wb5mmg/EnvTemperature
4	7	/stm32wb5mmg/Pressure
5	8	/stm32wb5mmg/BatteryVoltage
6	9	/stm32wb5mmg/RelativeHumidity
7	10	/stm32wb5mmg/AirQuality
8	11	/stm32wb5mmg/Co2
9	12	/stm32wb5mmg/Tvoc
10	13	/stm32wb5mmg/Luminosity
11	14	/stm32wb5mmg/WhiteLuminosity
12	15	/stm32wb5mmg/Plant/1/SoilMoisture
13	16	/stm32wb5mmg/Plant/1/Temperature
14	17	/stm32wb5mmg/Plant/2/Temperature
15	18	/stm32wb5mmg/Plant/2/SoilMoisture
16	19	/stm32wb5mmg/Plant/3/Temperature
17	20	/stm32wb5mmg/Plant/3/SoilMoisture
18	21	/stm32wb5mmg/LightState
19	22	/stm32wb5mmg/TankLevel