Shelly automatic garden watering system, using AccuWeather server, Vegetronix sensors and data server, and Pushsafer

by Josep Fargas

30.12.2023

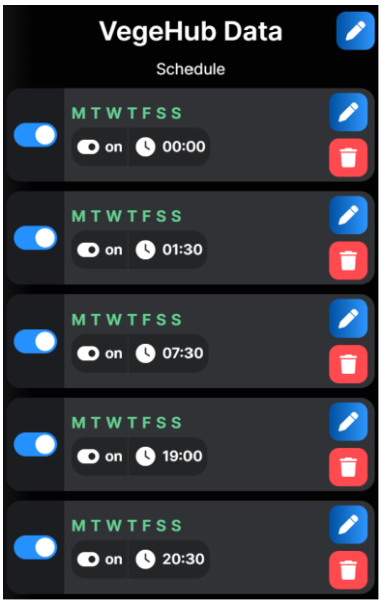
The following document describes the components of a Shelly system and script that controls 4 garden watering areas with different watering times duration and watering several times per day in summer. As explained below, the system connects to several servers and services.

The system is composed by the following 10 parts, which will be explained in detail:

1. A 6000L rainwater tank valve controlled by a Shelly Pro 4PM relay to fill the tank when needed,
2. A water pump valve controlled by the same previous Shelly Pro 4PM relay to water the garden,
3. 4 valves connected to 4 garden sprinkler watering circuits controlled by a second Shelly Pro 4PM relay,
4. A connection to AccuWeather server to read the current and forecast weather conditions to decide if watering is needed,
5. A connection to pushsafer.com to send phone notifications about the status of the system,
6. 3 sensors from Vegetronix.com, a moisture sensor, a water level sensor, and a water flow sensor, connected to the Vegetronix hub, which captures all the data and sends it to the VegeCloud.com server,
7. And a Shelly Pro 1 which holds the script and controls the whole system.
8. In addition, the Shelly App manages and controls the watering times and the schedule of the water filling times of the water tank.
9. Finally, as an alternative, the whole system can be monitored and controlled by Home Assistant as well.
10. Shelly Pro 1 script detailed description.

The following is a detailed explanation of all the components of the watering system:

1. The rainwater 6000L tank needs to be filled if there is not enough water for watering the garden. This needs to be done before watering the garden. The script reads the schedule of the Shelly Pro 1 and decides whether filling the tank or watering the garden, depending on the time minutes. For 0 minutes it fills the tank and for other minutes it waters the garden. The following picture on the left shows a typical Shelly App schedule where the relay checks the need to fill the tank at 00:00 and at 19:00, and waters the garden at 01:30, 07:30 and 20:30. The picture on the right shows the Shelly script configuration of the tank water levels and filling times.



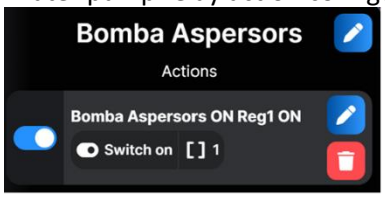

```
let configVegeHub = {
    RouteKey: "[redacted]",
    ReadApiKey: "[redacted]",
    Data: undefined,
    TankWaterLevel: undefined,
    TankWaterLevelMinLimit: 2000,
    TankWaterLevelLow: { volume: 3000, fillingTime: 60 },
    TankWaterLevelMedium: { volume: 3500, fillingTime: 45 },
    TankWaterLevelHigh: { volume: 4000, fillingTime: 30 },
    SoilMoisture: undefined,
    SoilMoistureThreshold: 6.36,
    WaterFlow: undefined,
    FillWaterTankMinutes: "00"
};
```

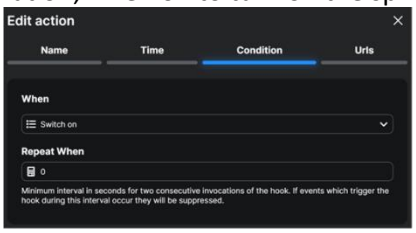
Depending on the water sensor level (low 3000L, medium 3500L or high 4000L, as shown above), the script fills the tank for 60, 45 or 30 minutes respectively. If the water level is lower than the minimum level of 2000L, the system will not water the garden and send a notification to warn the user.

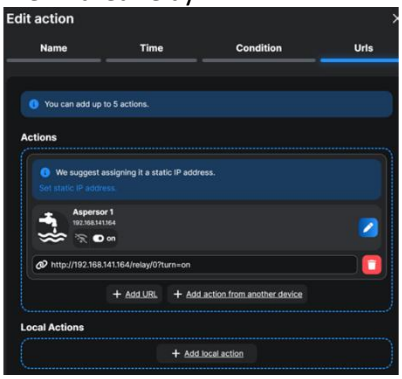
An additional interesting feature, not yet controlled by the script, is the “water tank cleaning filter valve” connected to a Shelly Pro 4PM, shown below in the pictures of points 8 and 9. After the rain, the filter needs to be cleaned for a minute because the tank is filled with rainwater collected from the house roof. This is done manually and might be added to another script in the future.

2. The water pump waters the different areas and needs to be activated when the sprinkler water areas valves are on. So, to minimize the script relay calls and script “timers”, which are limited, a cascade of calls allows the script to make just one call to start the whole system. This way, a water pump relay action activates the relay of the sprinkler 1 area. This procedure becomes very useful to add or remove watering areas without changing the script, just by changing actions in the Shelly App.
3. When the sprinkler 1 area turns off it turns on the sprinkler 2 area relay and so on, like in a daisy chain. When the sprinkler of the last area turns off, it turns off the water pump relay, as shown in the pictures below.

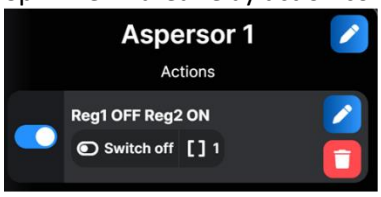
Water pump relay action configuration, when on to turn on the sprinkler 1 area relay:

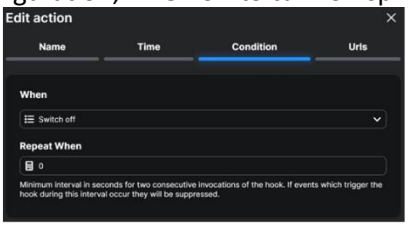


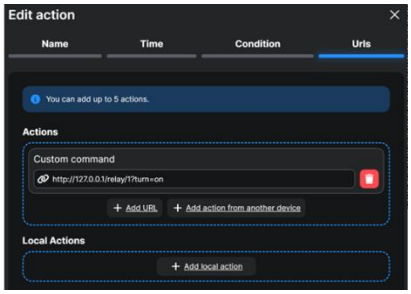




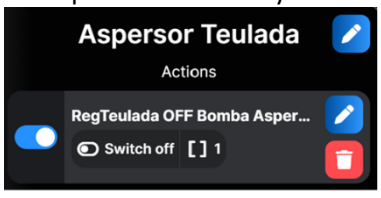
Sprinkler 1 area relay action configuration, when off to turn on sprinkler area 2:

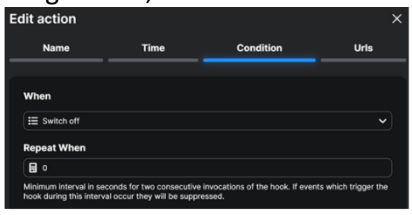


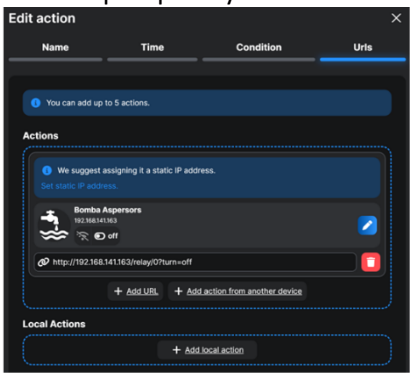


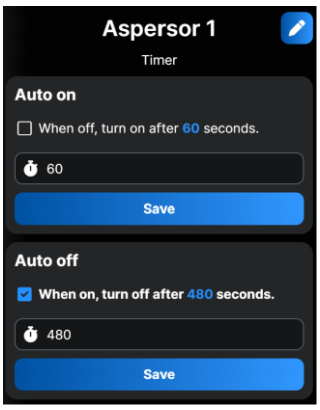


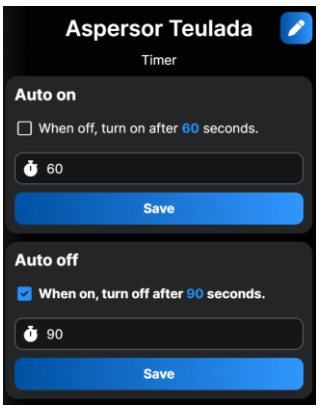
Last sprinkler area relay action configuration, when off to turn off the water pump relay:











Note that the Shelly App controls the watering time for the different sprinkler areas individually, via the "Auto off" timer feature. This way it is very easy to change the watering times without the need to edit the script.

The two pictures on the left show an example of the first and last sprinkler areas timers' configurations.

4. The AccuWeather server connection allows the system to know whether it needs to water the garden or not. Right now, the system checks the current conditions, especially for the night and day rain and precipitation of the last 6 hours and informs about the forecast. The image below shows the Shelly script configuration of the variables used for the AccuWeather connection:

```
let configAccuWeather = {
    APIKEY: "[redacted]",
    ForecastEndpoint: "http://dataservice.accuweather.com/forecasts/v1/daily/1day/",
    CurrentEndpoint: "http://dataservice.accuweather.com/currentconditions/v1/",
    Pobleta: {
        locationName: "Pobleta",
        locationKey: "92947_PC",
        currentCloudCover: undefined,
        currentHasRain: undefined,
        currentPrecipitationPast6HoursValue: undefined,
        currentPrecipitationPast6HoursValueThreshold: 5,
        currentPrecipitationPast6HoursUnit: undefined,
        forecastHasDayRain: undefined,
        forecastHasNightRain: undefined,
        forecastDayCloudCover: undefined,
        forecastDayRainProbability: undefined,
        forecastDayRainValue: undefined,
        forecastDayRainUnit: undefined,
        forecastNightCloudCover: undefined,
        forecastNightRainProbability: undefined,
        forecastNightRainValue: undefined,
        forecastNightRainUnit: undefined
    }
};
```

(For the script to work, you need to configure an AccuWeather account, this guide assumes it is already created).

5. The connection to pushsafer.com allows to send phone (and/or email) notifications about the status of the system, to inform the user about the watering system process, possible errors, and status. The following image shows the configuration of the pushsafer.com connection parameters:

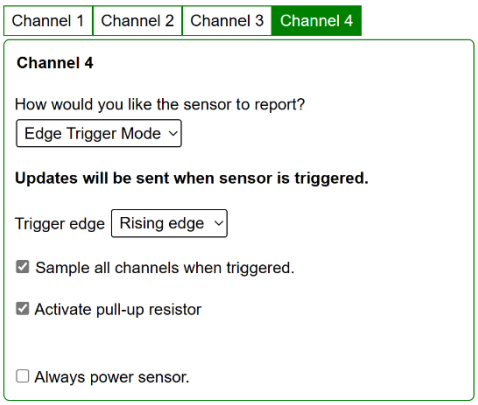
```
//pushsafer.com notifications parameters
let NotificationArgs = {
    sUrl: 'https://www.pushsafer.com/api',
    Key: '[redacted]',
    DeviceID: '[redacted]',
    Title: 'undefined',
    Message: 'undefined',
    Icon: '149',          //Tap
    Sound: '38',          //Beep short
    Vibration: 'blank'    //Default
};
```

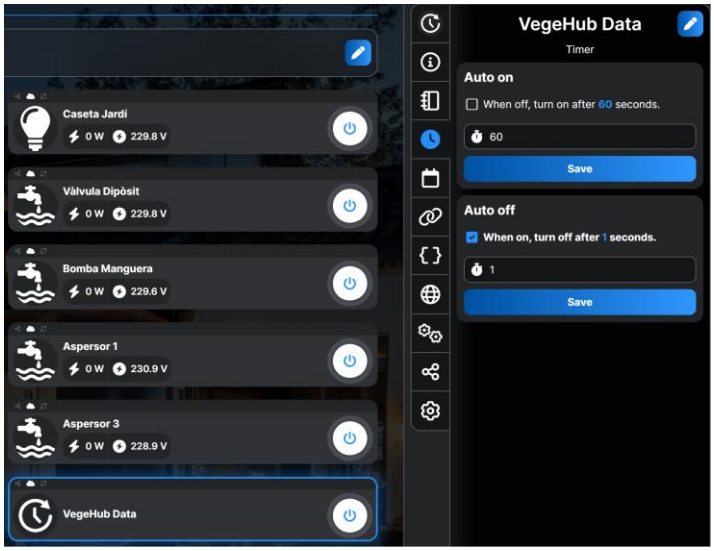
Note that you can not only send different notification messages, but you can also customize the icon, sound, and device vibration to more easily inform the user.
(For the script to work, you need to configure a pushsafer.com account, this guide assumes it is already created).

6. Vegetronix.com has different sensors especially designed to control the basic functionalities of such a watering system, like to control and monitor the water level of the 6000L tank -to make sure that there is enough water-, the soil moisture -to avoid watering the garden if there is enough moisture- and the water pump flow -to make sure that water is running through the pipes when the valves open. These sensors

connect to a hub which sends the data to a server, i.e. the free VegeCloud.com server. The hub sends updates to the server every 10 minutes with the data it collects from all the sensors every 5 minutes. In addition, the sensors' data needs to be updated just before watering or filling the tank, to make sure that the system uses the latest data. This is done by the Shelly Pro 1 as explained below.

Very interestingly, the Shelly Pro 1 connects to a channel of the hub, in this case channel 4, using its dry contact and closes the circuit for just a second to trigger the updates of the hub by configuring the "Auto off" timer in the Shelly App. As explained below in point 7, the script turns on the Shelly Pro 1 relay whenever it decides to read the sensors and launches the watering system. The image below on the left shows the Vegetronix hub channel configuration and, the one on the right, the Shelly Pro 1 timer configuration in the Shelly App.

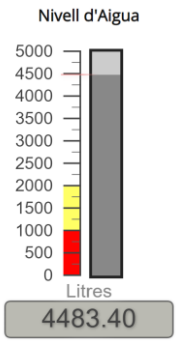




The other 3 channels of the hub are used for the 3 sensors, water level, soil moisture and water flow.

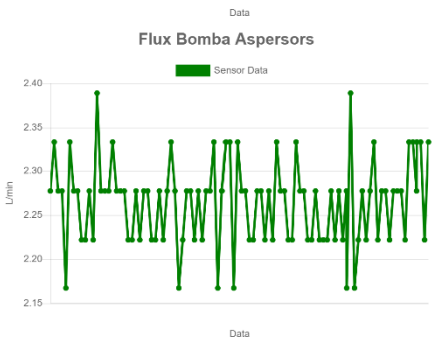
The following images correspond to the sensors' graphs generated by VegeCloud.com server:

Water level sensor reading



Soil moisture sensor readings

Water flow sensor readings



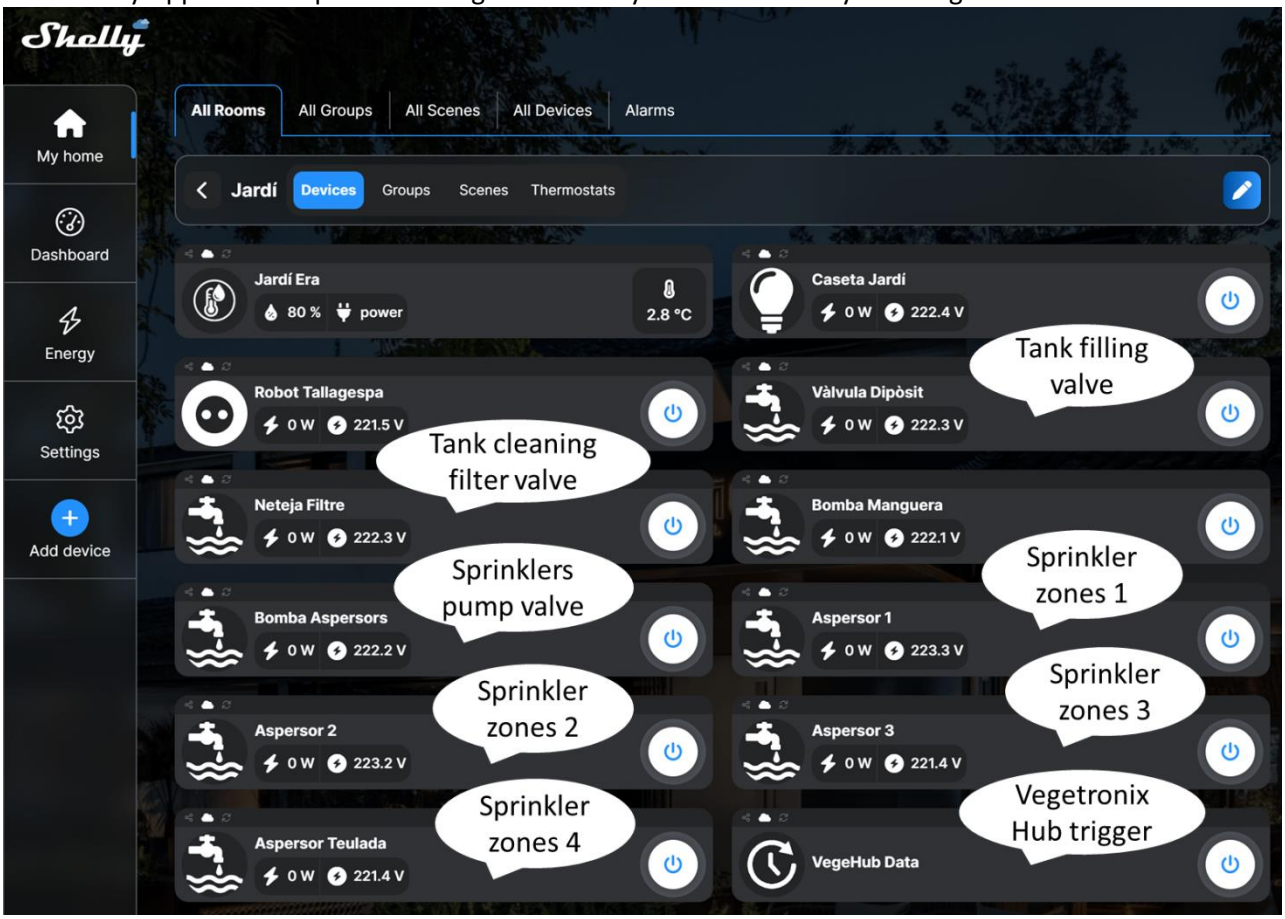
The purpose of this document is not to explain how to configure the Vegetronix sensors, nor the hub or the cloud server. The document explains how the script connects to the REST API of the server and how the Shelly Pro 1 triggers the updates of the hub, as explained below.

7. The main "heart" of the system is the Shelly Pro 1, called "VegeHub Data" in the App, which automatically manages and controls the whole system with a script, which triggers the Vegetronix hub to update the sensors dynamically in the server, as explained above, and configures the schedule of times to water the garden and fill the tank with water through the Shelly App (shown in the left picture of point 1 above).

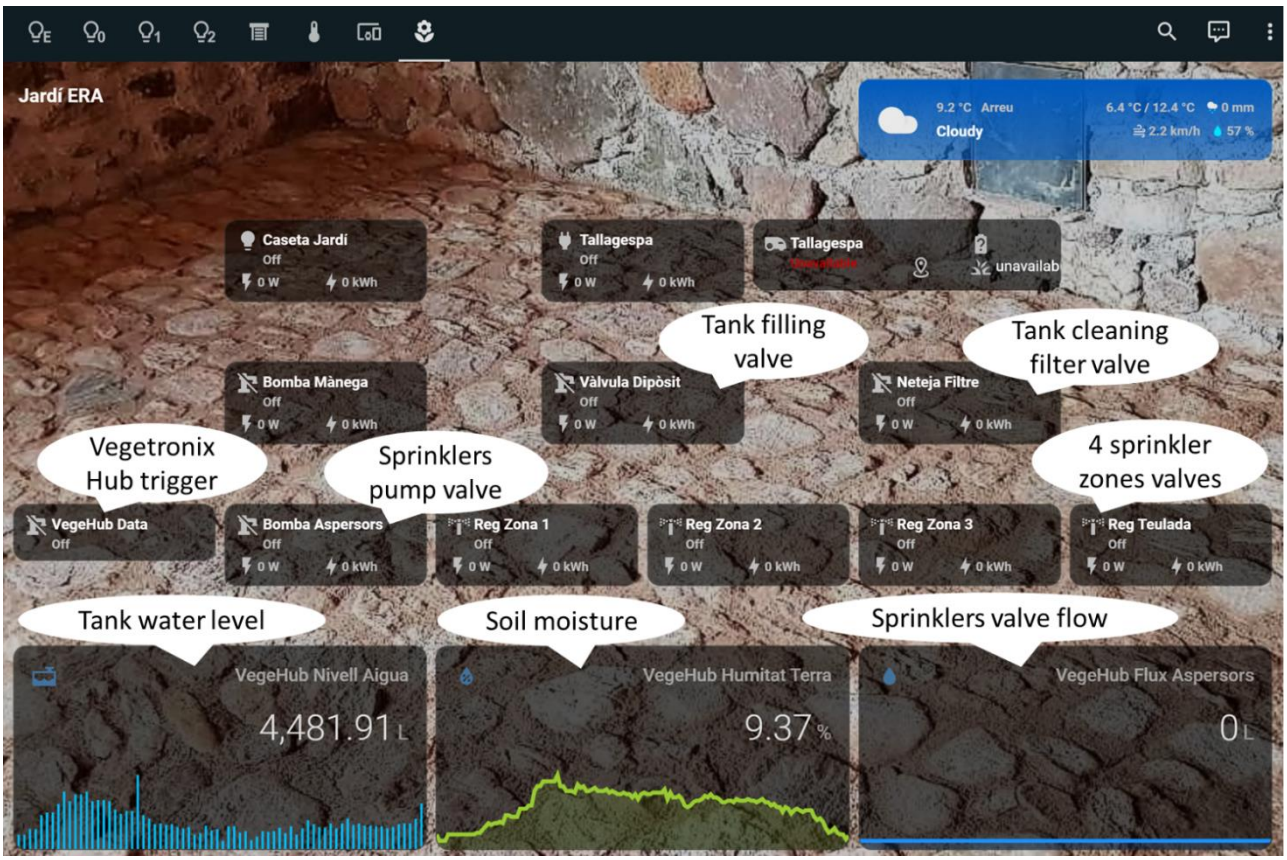
Point 10 will explain in detail the script run by this Shelly.

8. The Shelly App is also a fundamental part of the watering system because not only it configures all the relays, as shown above, and displays the whole system functionalities, but also and more importantly, it allows the user to configure the watering schedule without the need to open the script and program anything. This is fundamentally important because during the summer, and in function of the weather heat, you might need to water the garden more times per day and during a longer period of time.

   Main Shelly App screen capture showing all the Shelly devices and relays of the garden “room”:



9. As an alternative, the following image is a screen capture of the Home Assistant garden management integration:



Note how all the sensors are replicated from the Shelly App and seamlessly integrated with other devices, like the lights, also controlled by Shelly relays, and the grass mowing robot “Tallagespa”, shown as unavailable because it is turned off since it is winter. In addition, the graphs show how the VegeCloud.com server data is integrated in HASS, with a much more accessible, compacted and dynamic data than from the VegeCloud.com server.

10. Finally, the following paragraph explains how the Shelly Pro 1 script works and its main parts and functions. Note how the decision making process is separated from the sensors’ reading process and support functions, to make it easier to understand the flow of the program.

    The initial definitions of the script hold all the variables that need to be either changed by the user or filled by the script before making decisions.

    The “Shelly.addEventHandler” holds the main part of the script which calls the different functions:
    - “getLocationCurrentConditions” and “getLocationForecast”.
    - “getTimeNow” gets the time to compare it with the sensors’ one to make sure it is up to date.
    - Then a timer calls “HttpGetVegeHubData” to get the data about the 3 Vegetronix sensors.
    - Finally, a second timer calls “processActionToCarry” to decide what action to carry away.

    As you can see in the code, since the calls are asynchronous, the timers’ times of the two last functions need to be fine-tuned to allow the calls to occur one after the other.

    - Function “HttpGetVegeHubData” gets the VegeCloud data from the server. Note that to get the latest data, the query passed to the REST server, “?limit=10&order=desc&unix_timestamp=true”, is descended ordered and limited to 10 records, this ensures to get the latest data of the 3 sensors. It also gets the unix time stamp to be able to compare it to the one of the Shelly relay. Finally, note that at the end of the script, the function sets the “ActionToCarry” (it includes any possible error), and the notification title and message to inform the user. This function sets 3 possible values of “ActionToCarry”: “DoNothing”, “FillWaterTank” or “WaterGarden”, depending on the values of the 3 sensors. Note that this process takes a little bit long -around 45 seconds, so the timer is set to 50- because when the Shelly Pro 1 closes the

circuit it makes the Vegetronix hub to get the data from all the 3 sensors and then send it to the VegeCloud server, and finally, this needs to be ready to serve it to any client.

- Function “processActionToCarry” holds all the decisions of the script. The first thing it does is to check for errors. If there are no errors, it checks the weather conditions. Then it checks whether it needs to fill the water tank, first by checking the time (if the minutes are 0) and then by checking the water level. Finally, it checks the variable “ActionToCarry” defined by the previous function to see whether there is not enough water in the tank, or whether there is no need to water the garden (due to enough moisture or rain), or whether to go ahead and water the garden. Note how this function calls other functions like the notifications “HTTPPostNotification” and the relay actions “RelaySendCMD”, which allows to turn on or off the relay as well as using a timer.

- Functions “getVegeHubPortData”, “Volts2LitersTransformation”, “VH400Transformation” and “Volts2LitersMinuteTransformation” are used to get the port data and the transformations of volts to the different sensors’ units.

- Function “RelaySendCMD” sends a command to the relay using its IP, the channel and the command.

- Function “getTimeNow” gets the current time of the Shelly Pro 1.

- Function “HTTPPostNotification” sends the notification using pushsafer.com.

- Functions “getLocationCurrentConditions” and “getLocationForecast” get the current and forecast weather conditions from the AccuWeather server.

The following is a typical console.log message:

Turn auto off VegeHubData relay, check rain and updating VegeCloud server with latest data...	23:38:48
Waiting to read updated VegeHub data...	23:39:38
Data_1 = 1703975930 , Volts_1 = 2.588 V = 4431 L	23:39:41
Data_2 = 1703975930 , Volts_2 = 1.051 SM = 9.55 %VWC	23:39:41
Data_3 = 1703975930 , Volts_3 = 0.017 WF = 2 L/min	23:39:41
time.now = 23:38 , time.unixtime = 1703975928 , VegeHubUnixtime = 1703975930	23:39:41
Pobleta current clouds 30 %, current rain = false , precipitation past 6 hours 0 mm	23:39:46
Forecast day rain = false , forecast night rain = false	23:39:46
Pobleta forecast day clouds 39 %, day rain probability 3 %, day rain 0.0 "mm" , forecast night clouds 39 %, night rain probability 3 %, night rain 0.0 "mm"	23:39:46
No need to water the garden, enough soil moisture...	23:39:46
...End IntEraVegeHubData script...	23:39:46